

Auditor automático para nivel 2 del CMMI-SW

INGENIERIA INFORMÁTICA EN UNIVERSIDAD COMPLUTENSE DE MADRID



PROYECTO DE SISTEMAS INFORMÁTICOS, INGENIERÍA EN
INFORMÁTICA, FACULTAD DE INFORMÁTICA, UNIVERSIDAD
COMPLUTENSE DE MADRID

Giovanna da Cruz Botelho
Director: Celia Gutiérrez Cosío

Madrid, 6 de junio de 2013

Autorización de difusión y utilización

La abajo firmante, Giovanna da Cruz Botelho, matriculada en la asignatura Sistemas Informáticos de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autora, tanto la propia memoria, como el código, los contenidos audiovisuales incluso si incluyen imágenes de los autores, la documentación y el prototipo desarrollado durante el curso académico 2012-2013 bajo la dirección de la Profesora Celia Gutiérrez Cosío, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

Giovanna da Cruz Botelho

Dedicatoria

A mis padres por su interminable apoyo en todo momento de mi vida,
por sus enseñanzas, consejos y apoyo

A mis hermanos por su compañerismo, consejos y por estar conmigo
en tanto momentos importantes de mi vida

A mi novio que en los momentos difíciles estuvo ahí para ayudarme

A Profesora Celia Gutiérrez Cosío por darme la oportunidad
y por el tiempo que ha dedicado para leer este trabajo

Agradecimientos

Gracias a Profesora Celia Gutiérrez Cosío, profesora de la Universidad Complutense de Madrid, por su colaboración en todas las fases del proyecto, por la revisión de la memoria y por su disponibilidad, paciencia, apoyo y capacidad de guiar las ideas.

Índice

Índice de Ilustraciones	7
Índice de tablas	8
Resumen.....	9
1 Introducción	10
1.1 Estructura de la memoria.....	10
2 Descripción del problema	11
2.1 Elección de un modelo del CMMI	11
2.2 Clase del SCAMPI.....	11
2.3 Nivel de madurez	12
2.4 Definición de las evidencias	15
3 Metodología e instrumentación.....	17
3.1 Requisitos	17
3.1.1 Definición de las evidencias para cada práctica.....	17
3.1.2 Captura de datos	17
3.1.3 Diagnóstico.....	17
3.2 Diagrama de caso de uso	17
3.2.1 Insertar Datos en un proyecto	18
3.2.2 Quitar Datos de un proyecto.....	18
3.2.3 Pedir el Diagnóstico.....	19
4 Diseño de la solución.....	20
4.1 Entorno.....	20
4.1.1 Lenguaje	20
4.1.2 Gestión de base de datos	20
4.1.3 Conector	20
4.1.4 IDE	20
4.1.5 Hardware donde fue ejecutado	21
4.2 Simplificaciones	21
4.3 Evidencias.....	21
4.4 Base de Datos.....	31
4.5 Evaluación	32
4.6 Proyecto	33
4.7 Diagrama de clase	34
4.8 Interfaces.....	36

4.8.1 Interfaz Principal	36
4.8.2 Caja de Dialogo de los artefactos	38
4.8.3 Interfaz Diagnóstico	39
4.9 Ejemplos Ejecuciones	41
4.9.1 Primera Ejecución.....	41
4.9.2 Ejecución con evidencia en casi todas las áreas	46
5 Conclusiones y trabajo futuro	49
Bibliografia	50
Anexo 1: Instalación	51
Anexo2: Ejecución en Eclipse con MySQL.....	52

Índice de Ilustraciones

Ilustración 1: Comparación representaciones CMMI, extraída de [5]	12
Ilustración 2: Representación por etapas CMMI, extraída de [5]	13
Ilustración 3: Diagrama de caso de uso.....	18
Ilustración 4: Modelo de la Base de datos	31
Ilustración 5: Diagrama de Clase del proyecto.....	34
Ilustración 6: Interfaz Principal Prácticas Genéricas	36
Ilustración 7: Interfaz Principal Prácticas Específicas.....	37
Ilustración 8: Caja de Diálogo	38
Ilustración 9: Interfaz del Diagnóstico con las áreas no cumplidas	39
Ilustración 10: Interfaz del Diagnóstico con los procesos no cumplidos	40
Ilustración 11: Interfaz Inicial – Primera Ejecución	41
Ilustración 12: Caja Dialogo – Primera Ejecución.....	42
Ilustración 13: Caja Dialogo – Primera Ejecución, Inserta artefacto directo	42
Ilustración 14: Diagnóstico de la Primera Ejecución, Áreas no cumplidas.....	43
Ilustración 15: Diagnóstico de la Primera Ejecución, Procesos no cumplidos	43
Ilustración 16: Caja Dialogo – Primera Ejecución, Inserta artefacto indirecto	44
Ilustración 17: Diagnóstico de la Primera Ejecución, Áreas no cumplidas, después de inserción de evidencia	44
Ilustración 18: Diagnóstico de la Primera Ejecución, Procesos no cumplidos, después de inserción de evidencia.....	45
Ilustración 19: Pantalla Diagnóstico con las áreas no cumplidas para el caso citado.....	46
Ilustración 20: Pantalla Diagnóstico con los procesos no cumplidos para el caso citado.....	47
Ilustración 21: Caso dónde se alcanzó el nivel 2 de madurez	48

Índice de tablas

Tabla 1: Prácticas genéricas de nivel 2 extraída de [3]	21
Tabla 2: Área de proceso: Gestión de Acuerdos con Proveedores (SAM) extraída de [3]	23
Tabla 3: Área de proceso: Gestión de requerimientos (REQM) extraída de [3]	24
Tabla 4: Área de proceso: Planificación de Proyecto (PP) extraída de [3]	24
Tabla 5: Área de proceso: Monitorización y Control del Proyecto (PMC) extraída de [3]	26
Tabla 6: Área de Proceso: Gestión de Configuración (CM) extraída de [3]	28
Tabla 7: Área de proceso: Medición y Análisis (MA) extraída de [3]	29
Tabla 8: Área de proceso: Aseguramiento de la Calidad del Proceso y del Producto (PPQA) extraída de [3]	30

Resumen

Al seguir buenas prácticas con el objetivo de aumentar su nivel de madurez, una empresa que implementó el Modelo Integrado de Capacidad y Madurez (CMMI) debe pasar por una auditoría con el objetivo de recoger evidencias para comprobar si la organización ha alcanzado el nivel de madurez deseado.

El proyecto presentado en esta memoria consiste en el desarrollo de un sistema para auditar el cumplimiento de CMMI- SW nivel 2.

Palabras clave: CMMI, nivel de madurez 2, auditoría, proceso, evidencia, artefacto.

Abstract

By following best practices in order to increase his level of maturity, a company, which implemented the Model Capability Maturity Integrated (CMMI), must go through an audition with the aim of collecting evidence to see if the organization has reached the level of maturity desired.

The project presented in this report is the development of a system to audit the compliance of CMMI-SW Level 2.

Keywords: CMMI, maturity level 2, audition, process, evidence, artifacts.

1 Introducción

Alcanzar un nivel de madurez alto es un modo de ahorrar tiempo. Un departamento de una empresa de software que sigue buenas prácticas no va a cometer errores como hacer desde el principio un proyecto que ya está hecho en otro departamento, él va a reutilizar.

El término Modelo Integrado de Capacidad y Madurez - CMMI establece que la calidad de un producto va asociada a la calidad del proceso que se sigue para producirlo [1].

El CMMI define dos maneras de evaluar los procesos, por niveles de capacidad y por niveles de madurez. La más común es esta última, que permite la obtención de un nivel de madurez para la organización y es la que nos referiremos en este proyecto.

Una organización que desea alcanzar un nivel de madurez elevado puede usar las prácticas de CMMI e implementar las prácticas relacionadas a cada proceso necesario para el determinado nivel. Después, se puede hacer la certificación para garantizar que la empresa ha alcanzado el nivel de madurez deseado, entonces un auditor será el responsable por recoger las evidencias y, con esto, comprobar el nivel de madurez de la organización.

Para que el auditor pueda hacer su trabajo de forma más eficiente, se sugiere que el utilice una herramienta para guardar las evidencias encontradas. Después, para ayudar a obtener la información relevante sobre los procesos es importante proporcionar al auditor una herramienta que haga valoraciones rápidas de los procesos. Esta herramienta va a reducir la carga cognitiva asociada con la aplicación manual de esta actividad.

El software propuesto en ese trabajo de fin de grado guía el auditor en su evaluación y reporta las conclusiones alcanzadas a partir de los datos insertados por él.

1.1 Estructura de la memoria

Esta memoria se encuentra estructurada de la manera siguiente.

En el capítulo 2 se ofrece la descripción del problema. Se realiza un análisis del problema que se va a solucionar, estudiando cuál modelo se va elegir para el CMMI, cual clase de SCAMPI se adoptará, cual el nivel de madurez buscado y definiendo las evidencias.

En el capítulo 3 se modela la metodología utilizada: se definen los requisitos y el diagrama de casos de uso.

En el capítulo 4 hay el diseño de la solución con la descripción del entorno de desarrollo y del proyecto, enseñando la base de datos, el diagrama de clases y las interfaces. Además, hay ejemplos de ejecución del sistema.

Para finalizar, en el capítulo 5 hay las conclusiones y trabajos futuros.

En los anexos se puede ver detalles de la instalación y ejecución del sistema.

2 Descripción del problema

Para empezar a desarrollar la herramienta fue necesario primero analizar cuál sería el alcance del proyecto.

2.1 Elección de un modelo del CMMI

El CMMI (Modelo Integrado de Capacidad y Madurez) es un modelo desarrollado por el Instituto de Ingeniería del Software de la Universidad Carnegie Mellon, posterior al CMM (Modelo de Capacidad y Madurez).

“Un modelo de madurez y de capacidad (Capability Maturity Model, CMM), incluyendo CMMI, es una representación simplificada del mundo. Los CMMs contienen los elementos esenciales de los procesos eficaces. Estos elementos se basan en los conceptos desarrollados por Crosby, Deming, Juran y Humphrey.” [2]

El CMMI puede ser definido como:

“CMMI (Capability Maturity Model Integration) es un conjunto de herramientas que ayudan a una organización a mejorar sus procesos de desarrollo de productos y servicios, adquisiciones y mantenimiento. El modelo CMMI ha evolucionado el concepto del Modelo CMM establecido para la maduración de la capacidad de Software, a un nivel que permite el crecimiento continuo y la expansión del concepto CMM para múltiples disciplinas, tales como el Capability Maturity Model for Software (SW-CMM), el Systems Engineering Capability Mode (SECM), el Integrated Product Development Capability Maturity Model (IPD-CMM) y otros modelos de mejora de procesos.” [2]

El modelo CMM y el modelo CMMI se diferencian básicamente en que el primero se enfoca principalmente a las organizaciones o áreas de Tecnologías de información y el segundo es un modelo integrado y mejorado que se puede aplicar a un número mayor de organizaciones de diferentes sectores.

Hay 3 modelos donde el CMMI puede ser aplicado: Desarrollo, Adquisición y Servicios. Para cada modelo son determinados áreas, procesos y niveles con el objetivo de asegurar la calidad de procesos y productos.

En una primera fase fue determinado cual sería el área analizada y fue elegido el área Desarrollo, y por lo tanto el proyecto se centra en el modelo CMMI-SW.

El modelo CMMI-SW del Instituto de Ingeniería de Software es el modelo empleado por la industria mundial del software, como un estándar de facto, en la evaluación de la capacidad y madurez que tienen estas empresas para desarrollar software de alta calidad. [1]

2.2 Clase del SCAMPI

Los equipos de evaluación usan métodos conformes a los requerimientos del ARC (Appraisal Requirements for CMMI) para guiar su evaluación y reportar sus conclusiones.

El Standard CMMI Appraisal Method for Process Improvement (SCAMPI) es un método de evaluación que cumple todos los requerimientos ARC. [3]

Hay tres clases de métodos de evaluación en el SCAMPI: A, B y C

En función de su grado de adaptación y rigurosidad se distingue entre:

- SCAMPI-A: Es el más formal que mide la idoneidad, despliegue e institucionalización de los procesos. Es el necesario para poder obtener un certificado de un determinado nivel de madurez.

- SCAMPI-B: Permite evaluar la idoneidad y el grado de despliegue de los procesos, mediante entrevistas o revisión documental. Es recomendable para hacer auditorías de los procesos de una organización antes de afrontar el proceso de certificación con la evaluación formal.

- SCAMPI-C: Mide la idoneidad de los procesos, mediante entrevistas o revisión documental. Es el método idóneo para poder obtener una foto rápida del estado de los procesos en una organización para comenzar un programa de mejora de procesos. [4]

Había que elegir cual sería la clase del SCAMPI sería la más adecuada a este proyecto y fue elegido la clase C por su simplicidad, pues es más rápido, más sencillo, el tamaño del equipo de valoración es reducido y se necesita de pocos recursos.

2.3 Nivel de madurez

Certificar CMMI, de acuerdo al modelo CMMI-SW (Capability Maturity Model Integration for Software) [2], consiste en generar evidencia que muestre que el área de aseguramiento de calidad controla que la organización realmente aplica los procesos que dice aplicar, alcanzando así un determinado nivel de madurez.

El modelo de CMMI- SW puede seguir la representación continua, con niveles de capacidad, o la representación por etapas, con niveles de madurez. La tabla siguiente compara los cuatro niveles de capacidad con los cinco niveles de madurez.

<i>Nivel</i>	<i>Representación continua Niveles de capacidad</i>	<i>Representación por etapas Niveles de madurez</i>
Nivel 0	Incompleto	
Nivel 1	Realizado	Inicial
Nivel 2	Gestionado	Gestionado
Nivel 3	Definido	Definido
Nivel 4		Gestionado cuantitativamente
Nivel 5		En optimización

Ilustración 1: Comparación representaciones CMMI, extraída de [5]

“La representación continua se ocupa de seleccionar tanto un área de proceso particular a mejorar como el nivel de capacidad deseado para ese área de proceso. En este contexto, es importante conocer si un proceso se ha realizado o está incompleto. Por lo tanto, al punto de partida de la representación continua se le da el nombre de “Incompleto”. La representación por etapas se ocupa de seleccionar múltiples áreas de proceso a mejorar dentro de un nivel de madurez; no es su interés principal que los procesos individuales se realicen o estén incompletos. Por lo tanto, al punto de partida de la representación por etapas se le da el nombre de “Inicial”.” [5]

En este proyecto fue adoptado el modelo de representación por etapas que plantea 5 niveles de madurez. Cada nivel hay que cumplir determinadas áreas de proceso, que a su vez han que cumplir metas genéricas y metas específicas. Las metas específicas son las características únicas que deben estar presentes para satisfacer el área de proceso y, en contrapartida, las metas genéricas describen las características que deben estar presentes para institucionalizar los procesos que implementan un área de proceso. La siguiente figura representa el modelo de representación por etapas del CMMI.

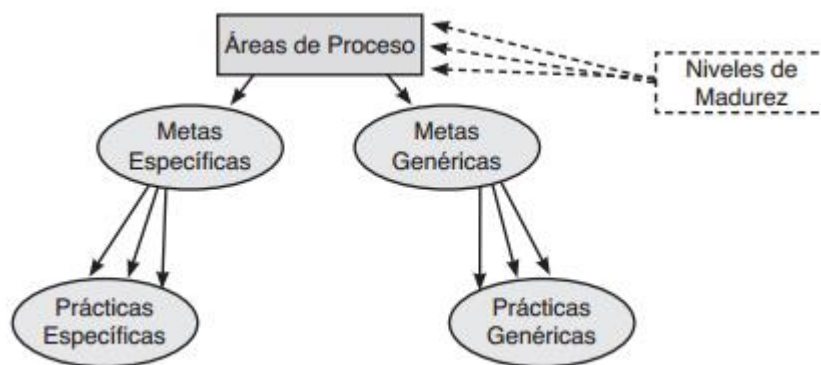


Ilustración 2: Representación por etapas CMMI, extraída de [5]

Cada nivel es un escalón bien definido de mejora de proceso y estabiliza una parte importante de los procesos de la organización:

- **NIVEL 1 - INICIAL:** Una organización CMMI Nivel 1 tiene procesos impredecibles, pobremente controlados y reactivos. El proceso de software se caracteriza como ad hoc y ocasionalmente caótico. Pocas actividades están definidas y el éxito de los proyectos depende del esfuerzo individual. Hay carencia de procedimientos formales, estimaciones de costos, planes del proyecto, mecanismos de administración para asegurar que los procedimientos se sigan.
- **NIVEL 2 - NIVEL ADMINISTRADO:** Una organización CMMI Nivel 2 asegura que sus procesos son planeados, documentados, realizados, monitoreados y controlados a nivel de proyectos. Los requerimientos, estándares y objetivos para los procesos, sus productos de trabajo y sus servicios son definidos y documentados. El estado de los productos es visible para la administración en puntos de control preestablecidos. En este nivel, los objetivos se cumplen con éxito, aún más, los objetivos de tiempo, calidad y costo para los procesos son satisfechos. [6] Las áreas de proceso de nivel 2 son: [3]

- Gestión de requerimientos (REQM)
- Planificación de proyecto (PP)
- Monitorización y control del proyecto (PMC)
- Gestión de acuerdos con proveedores (SAM)
- Gestión de configuración (CM)
- Aseguramiento de la calidad de proceso y producto (PPQA)
- Medición y Análisis (MA)

• **NIVEL 3 - NIVEL DEFINIDO:** Una organización CMMI Nivel 3 asegura que sus procesos son definidos. Un proceso definido es un proceso administrado que es ajustado desde el conjunto de estándares del proceso de la organización de acuerdo con las guías de ajuste; tiene una descripción del proceso mantenida, y aporta productos de trabajo, medidas y otra información de mejoramiento del proceso para los activos del proceso de la organización. Mientras que el nivel 2 se enfoca en los proyectos, el nivel 3 se enfoca en la definición de estándares a nivel organizacional. [6] Para alcanzar el nivel de madurez 3, es necesario implementar todas las áreas de proceso relativas al nivel 2, además de las siguientes áreas de proceso: [3]

- Desarrollo de requerimientos (RD)
- Solución técnica (TS)
- Integración de producto (PI)
- Verificación (VER)
- Validación (VAL)
- Definición de procesos de la organización (OPD)
- Enfoque de procesos de la organización (OPF)
- Formación organizativa (OT)
- Gestión integrada de proyecto (IPM)
- Gestión de riesgos (RSKM)
- Análisis de decisiones y resolución (DAR)

• **NIVEL 4 - NIVEL CUANTITATIVAMENTE ADMINISTRADO:** Una organización CMMI Nivel 4 asegura que sus procesos son controlados usando estadísticas y otras técnicas cuantitativas. Se establecen Objetivos Cuantitativos de Calidad y Performance, y son usados como criterios para

administrar los procesos. Las causas especiales de variación son identificadas y corregidas para evitar futuras ocurrencias. La diferencia esencial con el nivel 3 es que en este nivel la performance de los procesos es estadísticamente predecible. [6]Para alcanzar el nivel de madurez 4, es necesario implementar todas las áreas de proceso relativas al nivel 3 de madurez, además de las siguientes: [3]

- Gestión cuantitativa de proyecto (QPM)
- Rendimiento de procesos de la organización (OPP)

• NIVEL 5 - NIVEL OPTIMIZADO: Una organización CMMI Nivel 5 asegura que sus procesos son mejorados continuamente en base a un entendimiento de las causas comunes de variación. Se enfoca en la mejora continua de la performance a través de la incorporación de mejoras innovadoras en Tecnología y Proceso. La principal diferencia con el nivel 4 es el tipo de variación al que se apunta (variaciones comunes. [6]Para alcanzar el nivel 5 de madurez, es necesario implementar todas las áreas de proceso relativas al nivel de madurez 4, además de las siguientes: [3]

- Análisis causal y resolución (CAR)
- Gestión del rendimiento de la organización (OPM)

Fue elegido que el software iba decir si la organización había alcanzado el nivel 2 de madurez, es decir hay que estar implementados todos los procesos de las siguientes áreas: Gestión de requerimientos (REQM), Planificación de proyecto (PP), Motorización y control del proyecto (PMC), Gestión de acuerdos con proveedores (SAM), Gestión de configuración (CM), Aseguramiento de la calidad de proceso y producto (PPQA) y Medición y Análisis (MA)[Certificación CMMI-SW].

Alcanzar el nivel 2 de CMMI implica que la organización aplica procesos básicos de gestión de los proyectos como calendario, atas, registros, es decir, la organización planea sus procesos, los ejecuta, los mide y los controla.

2.4 Definición de las evidencias

Después de la definición del proyecto, el próximo paso sería definir cuáles serían las evidencias buscadas. Estas evidencias se les conoce como indicadores de implementación de la práctica (PII) y se distinguen tres tipos:

- Artefacto directo: salidas tangibles que resultan de la implementación directa de una práctica. Los artefactos directos pueden ser documentos, entregables, productos, material de formación, etc. [3]
- Artefacto indirecto: artefactos que son consecuencia de la implementación de una práctica, pero que no son el propósito para el cual se realiza la práctica. Los artefactos indirectos pueden ser actas de reunión, informes de estado, presentaciones, correos electrónicos, etc. [3]

- Afirmación: confirmaciones de palabra (entrevistas) o escritas que corroboran la implementación de una práctica específica o genérica. Ejemplos: entrevistas, tele conferencias, cuestionarios, etc. [3]

El área de desarrollo de CMMI contiene las áreas citadas y cada área contiene distintos procesos. El auditor debe analizar para cada proceso los artefactos directos, artefactos indirectos y afirmaciones existentes. Para decir cuáles serían los tipos de artefactos y afirmaciones buscados se basó en el Boletín de la ETSII ISSN: 2172-6620 (Escuela Técnica Superior de Ingeniería Informática Universidad Rey Juan Carlos) - "Guía práctica de supervivencia en una auditoria CMMI "(Numero 2011-002) [3].

Para considerar que hay una evidencia objetiva en un determinado proceso, hay que tener un artefacto directo y un artefacto indirecto o una afirmación, conforme la siguiente ecuación:

EVIDENCIA OBJETIVA = ARTEFACTO DIRECTO AND (ARTEFACTO INDIRECTO OR AFIRMACIÓN)

En resumen, el sistema permite que para cada proceso se guarde los artefactos directos, indirectos y afirmaciones relacionados y, después que el auditor rellene las evidencias encontradas, el software debe decir el diagnóstico, es decir, si la empresa ha alcanzado el nivel 2 de madurez o no.

3 Metodología e instrumentación

3.1 Requisitos

La aplicación consta de 3 módulos principales, que son clave para la terminación del proyecto. Los módulos principales son: “Definición de las evidencias para cada práctica”, “Captura de Datos” y “Hacer el Diagnóstico”.

3.1.1 Definición de las evidencias para cada práctica

Primero hay que seleccionar cuales son las evidencias para cada práctica, pues con eso se puede enseñar en la interfaz cuales son los artefactos directos, artefactos indirecto y afirmaciones que el auditor está buscando e irá insertar en las pantallas.

3.1.2 Captura de datos

La captura de datos es esencial para el sistema pues es a partir de esto que el sistema va a tomar decisiones.

3.1.3 Diagnóstico

El Diagnóstico es el propósito del sistema. En el diagnóstico debe mirar los datos que fueron capturados y decir si el proyecto ha alcanzado o no el nivel deseado.

3.2 Diagrama de caso de uso

A partir de los requisitos expuestos en el apartado anterior se han identificado los casos de uso correspondientes a la aplicación. A continuación se van a detallar mediante diagramas y explicaciones los casos de uso englobados por cada requisito.

Dentro del módulo Definición está implícito que es necesario un trabajo de definir los artefactos directos, artefactos indirectos y evidencias que van a estar en la interfaz. Esto no se va a poner como un caso de uso pues va a ser considerado como un prerequisite para la creación del sistema y no un caso de uso del sistema.

Dentro del módulo captura de datos, una vez que el usuario ha accedido a aplicación el puede realizar las siguientes acciones:

- Insertar Datos en un proyecto
- Quitar Datos de un proyecto

Dentro del módulo Diagnóstico, una vez que el usuario ha accedido a aplicación el puede realizar la siguiente acción:

- Pedir el Diagnóstico

Sigue el diagrama de casos de uso del sistema.

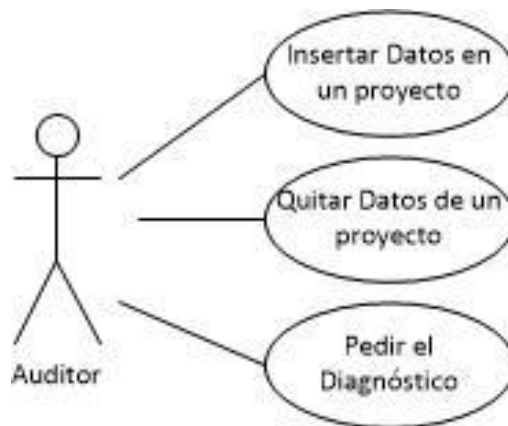


Ilustración 3: Diagrama de caso de uso

El diagrama de caso de uso arriba describe las acciones que el sistema debe permitir que el auditor pueda hacer.

3.2.1 Insertar Datos en un proyecto

Nombre: Insertar Datos en un proyecto.

ID: N/A.

Actor: Auditor.

Descripción breve: El auditor inserta artefactos directos, artefactos indirectos y/o afirmaciones en un proyecto.

3.2.2 Quitar Datos de un proyecto

Nombre: Quitar datos de un proyecto.

ID: N/A.

Actor: Auditor.

Descripción breve: El auditor quita artefactos directos, artefactos indirectos y/o afirmaciones en un proyecto.

3.2.3 Pedir el Diagnóstico

Nombre: Pedir el Diagnóstico

ID: N/A.

Actor: Auditor.

Descripción breve: El auditor pincha el botón Diagnóstico y el sistema enseña una pantalla con el resultado do diagnóstico, si el proyecto ha alcanzado o no el nivel 2 de madurez, y si no ha alcanzado enseña lo que hace falta.

4 Diseño de la solución

4.1 Entorno

Las herramientas utilizadas para desarrollar la aplicación son las siguientes.

4.1.1 Lenguaje

El lenguaje de programación base seleccionado ha sido Java y para la comunicación con la base de datos el SQL.

4.1.2 Gestión de base de datos

Se ha elegido MySQL (<http://dev.mysql.com/downloads/>) como sistema de gestión de bases de datos por ser el más utilizado en aplicaciones web (Wikipedia MySQL, 2013). Entre las razones de esta aceptación destaca que dispone de una herramienta visual de diseño, MySQL Workbench. Esta herramienta simplifica enormemente el diseño de las bases de datos gracias a la posibilidad de crear esquemas entidad relación y convertirlos directamente en tablas y relaciones (MySQL, 2013).

Para la base de datos, fue instalado un servicio de MySQL de la versión 5.5, el MySQL Community Server, que puede ser bajado de la página del MySQL: <http://www.mysql.com/>.

Para facilitar la creación de la base de datos fue instalado el aplicativo "MySQL Workbench" que aparece el siguiente nombre en la página de MYSQL: mysql-workbench-gpl-5.2.47-win32.msi.

4.1.3 Conector

Para permitir la conversación entre MySQL y Eclipse hay que bajar también un Conector JAVA llamado mysql-connector-java-5.1.24-bin, que se encuentra en la pasta "Lib" del código.

4.1.4 IDE

Se ha elegido el Eclipse como IDE." Eclipse es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse) (Wikipedia Eclipse, 2013).

4.1.5 Hardware donde fue ejecutado

El hardware ejecutado es un ordenador cuyo procesador es un intel core i7 y contiene 6 GB de memoria cuyo sistema operacional es el Windows 8, pero seguramente no es necesario un ordenador tan potente para la ejecución del software desarrollado en este proyecto.

4.2 Simplificaciones

Para que el proyecto fuera completo sería necesario que para cada área de proceso tuviera una tabla de prácticas genéricas y una tabla de prácticas específicas. Entonces, habría que definir los artefactos directos, artefactos indirectos y afirmaciones para cada práctica genérica y para cada práctica específica de cada área de proceso. Para simplificar el proyecto, fue definido que habría una interfaz dónde el auditor mete las prácticas genéricas y otras donde se pone las prácticas específicas para cada área de proceso, como en el “Guía práctica de supervivencia en una auditoría CMMI” [3]. La próxima sección explica como fueran elegidos cuáles serían los artefactos directos, artefactos indirectos y afirmaciones utilizados en este sistema.

4.3 Evidencias

Una Base de Datos de Indicadores de Implementación de la Práctica (Practice Implementation Indicators Data Base o PIIDB) es una base de datos que almacena evidencias de la implementación de las diferentes prácticas (SP y SG) definidas en CMMI. Las organizaciones pueden proporcionar una PIIDB como entrada a la evaluación (de hecho es muy recomendable trabajar desde el inicio con una PIIDB, para ordenar el trabajo y obtener una evaluación constante). Esta PIIDB mostrará la trazabilidad de las prácticas del modelo a las áreas de proceso correspondientes y a las evidencias objetivas usadas para verificar la implementación de la práctica. [3]

Normalmente, completar una PIIDB es una tarea extensa y como el propósito de este proyecto no es crear un PIIDB y si crear un sistema que verifica si hay evidencia a partir de un PIIDB existente, fue utilizado el PIIDB propuesto en la “Guía de Práctica de Supervivencia en una auditoría CMMI” de referencia [3].

Sigue abajo las tablas del PIIDB elegido:

- Prácticas genéricas

Tabla 1: Prácticas genéricas de nivel 2 extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
GP 2.1 Establecer una política de la organización	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre	Comunicación del plan de calidad a la empresa (correos, wiki, página Web, etc.).

	firmado y respaldado por la gerencia.	
GP 2.2 Planificar el proceso	Documentos para planificar cada uno de los procesos, y que contienen su descripción, sus objetivos, sus responsabilidades, sus dependencias, las mediciones a realizar para el proceso, etc.	Horas imputadas al proyecto.
GP 2.3 Proporcionar recursos	Descripción de los equipos y recursos (humanos y materiales) disponibles para la realización del proceso.	Facturas de compra de equipos o de herramientas. Horas imputadas a las tareas de gestión del proyecto.
GP 2.4 Asignar responsabilidad	Documento de roles y responsabilidades para cada proceso.	Actas de reunión convocadas en las que se nombran los responsables.
GP 2.5 Formar al personal	Tareas de formación realizadas: temario de los mismos, materiales, objetivos de la realización de la formación, exámenes, etc.	Informe de asistencia a los cursos de formación por parte del personal.
GP 2.6 Gestionar configuraciones	Gestor de configuración del código fuente (SVN, CVS, etc.). Gestor de configuración documental (SharePoint, wiki, etc.). Sistemas de copias de seguridad.	Logs que muestren el uso de cada una de las herramientas de gestión de configuración.
GP 2.7 Identificar e involucrar a las partes interesadas relevantes	Plan de comunicación establecido en el que se identifican a los implicados en el proceso.	Actas de reunión en la que participan los diferentes involucrados en el proyecto. Correos de convocatoria.
GP 2.8 Monitorizar y controlar el proceso	Informes de medición intermedios de los productos software. Informes de medición del rendimiento de los procesos.	Acciones correctivas asociadas a las mediciones del rendimiento de los procesos. Comunicación de los resultados.
GP 2.9 Evaluar objetivamente la adherencia	Informes de auditoría interna y externa de los procesos.	Horas imputadas a tareas de auditoría. Registros de auditoría, contratación de auditores, etc.
GP 2.10 Revisar el estado con el nivel directivo	Resultado de la reunión con la dirección para revisar los procesos de la organización.	Actas de reunión, convocatorias, calendario de planificación, etc.

- Gestión de Acuerdos con Proveedores (SAM)

Tabla 2: Área de proceso: Gestión de Acuerdos con Proveedores (SAM) extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
SG 1 Establecer los acuerdos con proveedores		
SP 1.1 Determinar el tipo de compra	Política de acuerdos con proveedores, definiendo los tipos de compras posibles (productos off-the-shelf, productos a medida, etc.).	Requisito del proyecto donde se describe el módulo a contratar.
SP 1.2 Seleccionar los proveedores	Plantilla de homologación de proveedores. Listado de proveedores.	Informe de homologación.
SP 1.3 Establecer los acuerdos con el proveedor	Contrato con el proveedor.	Acta de reunión donde se ha realizado el acuerdo.
SG 2 Satisfacer los acuerdos del proveedor		
SP 2.1 Realizar el acuerdo del proveedor	Informes de cierre de acuerdos y de progreso del proveedor.	Incidencias registradas. Actas de reuniones de seguimiento.
SP 2.2 Monitorizar los procesos seleccionados del proveedor	Informes de seguimiento del proveedor. Informes de discrepancias.	Horas imputadas a la monitorización de actividades del proveedor.
SP 2.3 Evaluar los productos a medida seleccionados del proveedor	Listado de productos de trabajo seleccionados a evaluar del proveedor e informes sobre la selección.	Horas imputadas a la evaluación de productos de terceros.
SP 2.4 Aceptar los productos adquiridos	Procedimiento y resultados de pruebas de aceptación.	Incidencias históricas y resolución.
SP 2.5 Transferir los productos	Planes de transición y despliegue, gestión del cambio, paso a producción, a pre-producción, etc. Informes de formación sobre los nuevos productos.	Horas imputadas por cada empleado involucrado a actividades de formación relacionadas con el producto. Informe de incidencias durante el despliegue.

- Gestión de requerimientos (REQM)

Tabla 3: Área de proceso: Gestión de requerimientos (REQM) extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
SG 1 Gestionar los requerimientos		
SP 1.1 Obtener una comprensión de los requerimientos	Aplicación de un listado de criterios definidos para la evaluación y la aceptación de los requisitos. Resultados del análisis de los requisitos frente a los criterios de aceptación.	Correos electrónicos o actas de reunión en los cuales se evidencia el entendimiento, negociación o cambio de requisitos.
SP 1.2 Obtener el compromiso sobre los requerimientos	Documento de requisitos aceptado. Acta de reunión donde se aceptan los requisitos.	Histórico de requisitos cuyo estado pasa a ser aceptado.
SP 1.3 Gestionar los cambios en los requerimientos	Peticiones de cambio asociadas con requisitos. Requerimientos versionados. Tareas para cada petición de cambio: trazabilidad de la petición de cambio con las tareas.	Histórico de requisitos cuyo estado haya pasado a “abierto” tras haber sido “cerrado”. Estimación de las tareas asociadas al cambio en un requisito.
SP 1.4 Mantener una trazabilidad bidireccional de los requerimientos	Matriz de trazabilidad entre requisitos y los demás elementos que componen el producto software (ej. diseño, casos de prueba, código fuente, etc.).	Análisis de cambio donde se ha utilizado la matriz de trazabilidad para valorar el impacto.
SP 1.5 Identificar las inconsistencias entre el trabajo del proyecto y los requerimientos	Informe de pruebas. Listado de inconsistencias.	Acciones correctivas asociadas a las inconsistencias encontradas entre el proyecto y los requisitos.

- Planificación de Proyecto (PP)

Tabla 4: Área de proceso: Planificación de Proyecto (PP) extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
SG 1 Establecer estimaciones		
SP 1.1 Estimar el alcance del proyecto	Oferta o plan de proyecto donde se indican el alcance del sistema. Descripción de las tareas a realizar durante el proyecto.	Horas imputadas a la tarea de estimación del alcance, oferta inicial, estudio de viabilidad, etc.
SP 1.2 Establecer las estimaciones de los atributos del producto de trabajo y de las tareas	Diagrama de Gantt en el que se describen la duración de las tareas, en base a una estimación por analogía.	Horas imputadas a la realización de la estimación por analogía, punto función, puntos póker, etc.

	Planificación del sprint backlog. Informe con los resultados de la estimación.	
SP 1.3 Definir el ciclo de vida del proyecto	Una sección, usualmente incorporada al plan de proyecto donde se describen las fases que contendrá el proyecto (análisis, diseño, despliegue, iteraciones, etc.), la relación entre estas fases y su ordenación temporal (cascada, iterativo, incremental, etc.).	Hitos definidos en un diagrama de Gantt.
SP 1.4 Determinar las estimaciones de esfuerzo y costes	Informe en el que se representan los resultados de la estimación del esfuerzo necesario y el método usado. Hoja de costes para el proyecto y el procedimiento de cálculo. Definición de recursos necesarios (memoria, capacidad de red, etc.) para la realización del proyecto.	Horas imputadas a la tarea de estimación. Herramienta de cálculo de esfuerzo y coste completada.
SG 2 Desarrollar un plan de proyecto		
SP 2.1 Establecer el presupuesto y el calendario	Sección de presupuesto del documento del plan de Proyecto. Diagrama de PERT en el que se identifican las distintas tareas y sus dependencias.	Horas imputadas a la tarea de elaboración del presupuesto y calendario. Herramienta de cálculo de esfuerzo y coste completada.
SP 2.2 Identificar los riesgos del proyecto	Matriz de riesgos identificados para el proyecto. Checklist que evalúa los riesgos para el proyecto.	Catálogo genérico de riesgos. Horas imputadas a la tarea de identificación de riesgos.
SP 2.3 Planificar la gestión de los datos	Listado de los datos gestionados en el proyecto, con la descripción del formato, requisitos de privacidad y seguridad. Descripción del sistema de Backup. Datos que requieren confidencialidad.	Estructura de carpetas y permisos para controlar datos confidenciales. Logs de backups realizados para el proyecto.
SP 2.4 Planificar los recursos del proyecto	Listado de equipamiento, instalaciones y software asociado con el proyecto. Listado de recursos humanos necesarios.	Orden de compra de equipamiento. Acta de reunión interna de arranque.
SP 2.5 Planificar el conocimiento y las habilidades necesarias	Listado de habilidades necesarias por parte de los miembros del equipo.	Actividades de formación para los perfiles del proyecto. Material de formación.

	Plan de personal y de nuevas contrataciones.	
SP 2.6 Planificar la involucración de las partes interesadas	Listado de los participantes del proyecto y rol que juegan en el mismo. Comunicación formal a las personas que participarán en el proyecto (cliente, desarrolladores, equipo de pruebas, etc.). Plan de comunicación y relaciones entre los participantes. Plan de proyecto.	Acta de reunión de arranque del proyecto en la que participan tanto el cliente como los principales involucrados en el proyecto y se explica el plan de comunicación.
SP 2.7 Establecer el plan de proyecto		Horas imputadas a la tarea de planificación. Plantilla de plan de proyecto.
SG 3 Obtener el compromiso con el plan		
SP 3.1 Revisar los planes que afectan al proyecto	Matriz de relaciones entre proyectos, planificación de proyectos y recursos asignados en la unidad organizacional. Documento con la ocupación de recursos de la organización.	Registro de resolución de conflictos (correo, acta, etc.).
SP 3.2 Reconciliar los niveles de trabajo y de recursos	Presupuestos renegociados. Control de la asignación y capacidad de los recursos Reestimación de las tareas de los implicados que tengan una dedicación que no sea aceptable.	Revisión en herramienta de gestión de personal y control de horas, así como el ajuste de recursos y tareas.
SP 3.3 Obtener el compromiso con el plan	Aceptación por los afectados del plan de proyecto.	Acta de reunión de inicio del proyecto con la participación del equipo.

- Monitorización y Control del Proyecto (PMC)

Tabla 5: Área de proceso: Monitorización y Control del Proyecto (PMC) extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
SG 1 Monitorizar el proyecto frente al plan		
SP 1.1 Monitorizar los parámetros de planificación del proyecto	Actas de las reuniones de seguimiento llevadas a cabo. Herramienta de seguimiento (por ejemplo Gantt de seguimiento, Trac, etc.). Identificación de desviaciones en el proyecto.	Registro de la revisión de las horas imputadas al proyecto.

SP 1.2 Monitorizar los compromisos	Actas de reunión de seguimiento, informes de avance, de cumplimiento de hitos, etc.	Horas imputadas al seguimiento del proyecto.
SP 1.3 Monitorizar los riesgos del proyecto	Histórico de cambios en los riesgos.	Acta de reunión de seguimiento en la que se tratan explícitamente los riesgos del proyecto.
SP 1.4 Monitorizar la gestión de datos	Servidor de integración continua. Registro de tareas de gestión de datos.	Logs del sistema de backups. Histórico de revisiones en gestor de configuración.
SP 1.5 Monitorizar la involucración de las partes interesadas	Actas de reunión de entrega de hitos intermedios.	Correos o notificaciones entre los implicados.
SP 1.6 Llevar a cabo revisiones de progreso	Informes de avance del seguimiento. Actas de reunión de seguimiento.	Horas imputadas por parte del equipo a tareas del proyecto. Impedimentos detectados durante las reuniones de seguimiento.
SP 1.7 Llevar a cabo revisiones de hitos	Actas de reunión de entrega intermedia, reuniones intermedias de chequeo con el cliente. Acta de reunión de final de un Sprint.	Histórico de entregables. Histórico de incidencias. Riesgos del proyecto revisados durante la realización del hito.
SG 2 Gestionar las acciones correctivas hasta su cierre		
SP 2.1 Analizar problemas	Incidencias registradas y analizadas.	Peticiones de cambio asociadas. Planificación modificada incluyendo las incidencias y su estimación.
SP 2.2 Llevar a cabo las acciones correctivas	Documento o registro de acciones correctivas.	Incidencias resueltas. Histórico de cambios de estado de las incidencias
SP 2.3 Gestionar las acciones correctivas	Histórico de acciones correctivas, participantes, planes derivados, etc.	Acta de reunión al final del hito en la que se incluye la revisión de las incidencias y las acciones correctivas asociadas.

- Gestión de Configuración (CM)

Tabla 6: Área de Proceso: Gestión de Configuración (CM) extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
SG 1 Establecer líneas base		
SP 1.1 Identificar elementos de configuración	Documento o herramienta donde se identifican los elementos de configuración de las líneas base. Pueden ser productos que se entregan al cliente, herramientas, diseños, planes de pruebas, prototipos, resultados de pruebas, documentos, etc.	Tiempo dedicado a su elaboración, horas, actas, etc.
SP 1.2 Establecer un sistema de gestión de la configuración	Herramienta de gestión de la configuración (SVN, CVS, etc.).	Histórico de revisiones y roles de acceso al sistema de gestión de la configuración. Logs de herramientas de gestión de configuración.
SP 1.3 Crear o liberar líneas base	Descripción de las entregas formales a realizar durante el proyecto, tanto de productos software como de documentación, describiendo los elementos que contiene.	Histórico de entregas formales realizadas.
SG 2 Seguir y controlar los cambios		
SP 2.1 Seguir las peticiones de cambio	Peticiones de cambio realizadas durante el proyecto.	Registro con la aprobación o denegación de un cambio en el sistema.
SP 2.2 Controlar los elementos de configuración	Servidor de integración continua que integra periódicamente el código realizado hasta ese momento, identificando errores o conflictos. No conformidades identificadas durante las auditorías internas y externas.	Logs de ejecuciones del servidor de integración continua. Registros de auditoría interna o externa.
SG 3 Establecer la integridad		
SP 3.1 Establecer registros de gestión de la configuración	Revisiones de las tareas de gestión de configuración Revisiones de los cambios implementados entre dos versiones de la línea base.	Incidencias en la gestión de configuración (ej. establecimiento de ramas, “merges” que no funcionan). Histórico de cambios en la herramienta de gestión de la configuración.
SP 3.2 Realizar auditorías de configuración	Informe de auditoría interna o externa.	No conformidades extraídas de la realización de la auditoría.

- Medición y Análisis (MA)

Tabla 7: Área de proceso: Medición y Análisis (MA) extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
SG 1 Alinear las actividades de medición y análisis		
SP 1.1 Establecer los objetivos de medición	Documento con los objetivos de medición, donde se indican los objetivos de negocio y su relación con los indicadores de medición.	Correos de sugerencias de indicadores. Histórico de indicadores.
SP 1.2 Especificar las medidas	Descripción de los indicadores de medición: unidades de medida, mecanismo de recogida, periodicidad de la recolección, objetivo de la medición, etc.	Actas de reunión para la definición de los indicadores. Histórico de resultados de las mediciones. Catálogo genérico de métricas.
SP 1.3 Especificar los procedimientos de recogida y de almacenamiento de datos	Descripción de los indicadores de medición: unidades de medida, mecanismo de recogida, etc. Sección en la documentación donde se indica cómo se almacenan los datos.	Procedimiento de medición y análisis desarrollado. Logs de las herramientas de recolección automática de datos.
SP 1.4 Especificar los procedimientos de análisis	Descripción de los indicadores de medición, umbrales y análisis a realizar.	Plantilla de los informes de indicadores.
SG 2 Proporcionar los resultados de la medición		
SP 2.1 Recoger los datos de la medición	Informe que contenga los datos extraídos de la medición.	Horas imputadas a realizar el informe. Logs de las herramientas de recolección automática de datos.
SP 2.2 Analizar los datos de la medición	Informe de análisis de los datos obtenidos.	Acta de reunión de análisis de datos. Acciones correctivas asociadas con el análisis.
SP 2.3 Almacenamiento de datos y los resultados	BBDD de indicadores, con los resultados de las mediciones anteriores y actuales.	Horas imputadas al análisis y almacenamiento de los resultados.
SP 2.4 Comunicar los resultados	Correo electrónico, acta, etc. donde se evidencie la comunicación de los resultados	Contestaciones a las comunicaciones de los resultados. Acciones correctivas identificadas en base a los resultados.

- Aseguramiento de la Calidad del Proceso y del Producto (PPQA)

Tabla 8: Área de proceso: Aseguramiento de la Calidad del Proceso y del Producto (PPQA) extraída de [3]

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO
SG 1 Evaluar objetivamente los procesos y los productos de trabajo		
SP 1.1 Evaluar objetivamente los procesos	Plan de calidad donde se han registrado las diferentes auditorías independientes que se realizarán a los proyectos.	Actas de reunión de evaluación de los procesos. Horas imputadas a las auditorías. Registro de auditoría.
SP 1.2 Evaluar objetivamente los productos y los servicios	Informes de pruebas de los productos y servicios. Informes de auditoría interna o externa realizada al proyecto.	Histórico de casos de prueba. Horas imputadas a las auditorías.
SG 2 Proporcionar una visión objetiva		
SP 2.1 Comunicar y asegurar la resolución de las no-conformidades	No conformidades detectadas durante la auditoría comunicadas a los proyectos y asignadas al responsable de la resolución.	Acciones correctivas asociadas a las no conformidades.
SP 2.2 Establecer registros	Plan de calidad e informes de auditoría. Almacenamiento de las no conformidades identificadas en las auditorías.	Horas imputadas a las auditorías.

4.4 Base de Datos

Fue implementada una base de datos siguiendo el modelo de base de datos relacional.

La base de datos contiene las siguientes tablas:

- Proyecto: Representa el proyecto, dónde el auditor va a insertar datos
- Área: Representa el área de proceso
- Practica: Representa las prácticas de proceso que deberán ser cumplidas. Una práctica contiene un artefacto directo, un artefacto indirecto y una afirmación, que van a ser utilizados en la evaluación.

Como se nota en el modelo de base de datos, un proyecto contiene determinadas áreas de proceso, las cuales contienen determinadas prácticas.

La captura de datos está hecha de forma que cada vez que el auditor mete nuevos datos en un proceso de un proyecto y elige la opción “Guardar”, se actualizan los datos del determinado proyecto.

Sigue el modelo de la base de datos:

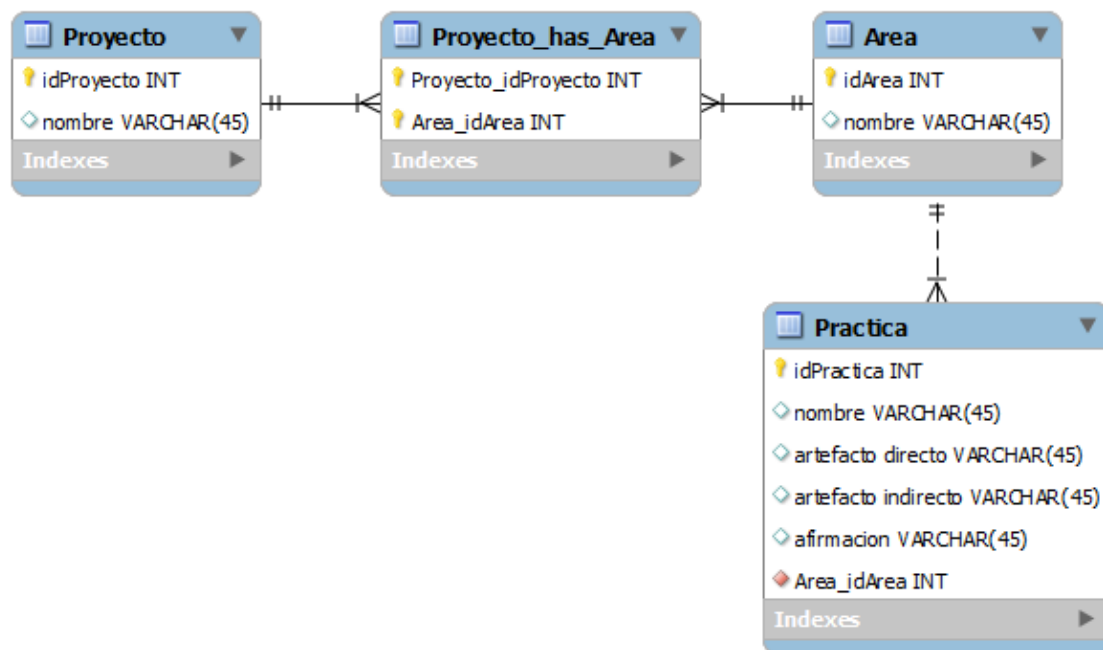


Ilustración 4: Modelo de la Base de datos

4.5 Evaluación

Para que una organización pueda alcanzar el nivel de madurez 2, es necesario que se implementen las áreas de proceso que CMMI define para ese nivel de madurez. Fue definido que se una organización implementa todas las áreas citadas en las tablas enseñadas arriba, entonces ha alcanzado el nivel 2 de madurez.

Para que un área de proceso sea correctamente implementada, deben alcanzarse los objetivos definidos para esa área de proceso, que a su vez se consiguen mediante la implementación de las prácticas de cada objetivo específicas y genéricas. Como fue dicho anteriormente, se consideró apenas las prácticas específicas para cada área de proceso y para las prácticas específicas se utilizó las prácticas definida en la tabla “Tabla1: Prácticas genéricas de nivel 2 extraída de [3]”. Para las prácticas específicas se utilizó las demás tablas.

Como fue dicho anteriormente, para considerar que hay una evidencia objetiva en un determinado proceso, hay que tener un artefacto directo y un artefacto indirecto o una afirmación, conforme la siguiente ecuación:

EVIDENCIA OBJETIVA = ARTEFACTO DIRECTO AND (ARTEFACTO INDIRECTO OR AFIRMACIÓN)

Siendo así, la evaluación va a determinar si hay evidencia para todas áreas y si hay, la organización ha alcanzado el nivel 2 de madurez y, en caso contrario, la organización no ha alcanzado el nivel 2 de madurez.

4.6 Proyecto

El proyecto fue diseñado de forma a separar la base de datos, las interfaces y el diagnóstico, así él se hace más flexible en caso de futuras mejoras y adaptaciones.

El proyecto está dividido en cuatro paquetes:

- Default: Contiene el Main
 - Main: La clase principal del proyecto, donde se inicia la ejecución.
- DataBase: Contiene las clases relacionadas a la base de datos
 - ComandosSql: Una clase donde están los comandos SQL usados por otras clases
 - Conexion: La clase responsable por la conexión con la base de datos
 - InicializaBD: La clase que inicia la base de datos.
 - ReiniciaBD: La clase que reinicia la base de datos.
- GUI: Contiene las clases relacionadas a las interfaces
 - MainFrame: La interfaz Principal y el JDialog donde el auditor inserta los artefactos.
 - UIDiagnostico: La interfaz con el diagnóstico de los proyectos.
- Model: Contiene las clases relacionadas al Diagnóstico
 - Diagnostico: La clase que haz la análisis de los artefactos y dice si ha alcanzado o no el nivel de madurez.

4.7 Diagrama de clase

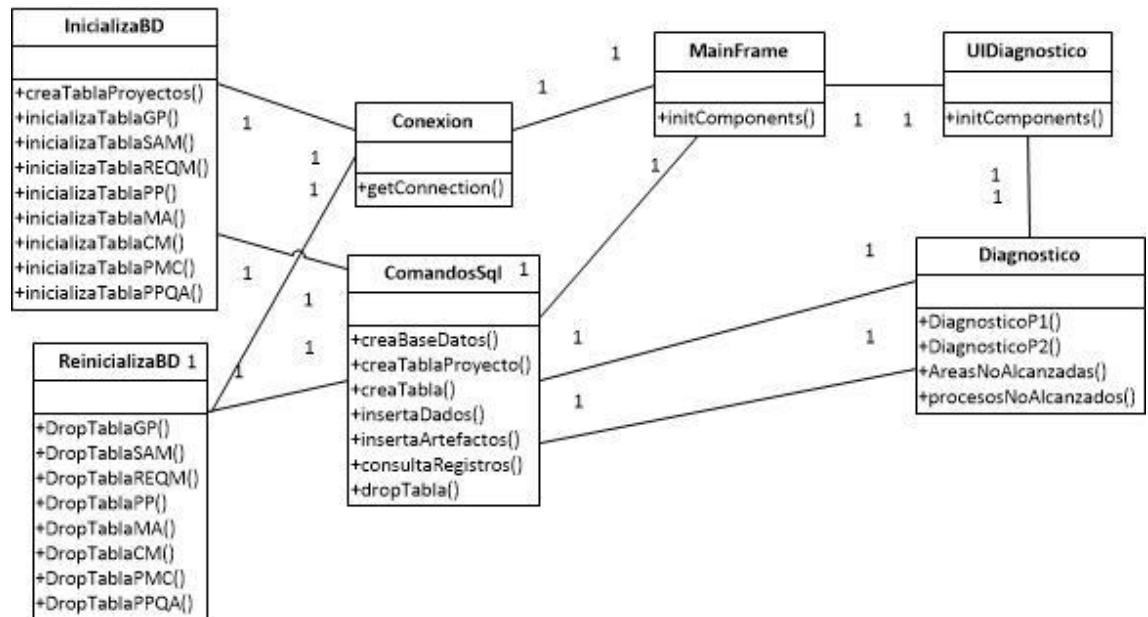


Ilustración 5: Diagrama de Clase del proyecto

– MainFrame: Esta clase contiene la interfaz principal del sistema y los métodos llamados al pinchar los botones de la interfaz, por esto hay que tener un elemento de la clase Conexión y un de la clase ComandosSql que son utilizados por métodos que hacen modificaciones en la base de datos.

Contiene una instancia de la clase UIDiagnostico que es llamado cuando se desea obtener el Diagnostico.

– UIDiagnostico: La interfaz con el diagnóstico de los proyectos.

Contiene una instancia de la clase Diagnostico que es llamada cuando se pide el diagnóstico.

– ComandosSql: Esta clase contiene los comandos SQL usados por otras clases.

Fue criada para evitar la repetición de código y simplificar el uso de las funciones que hacen inserciones, actualizaciones, exclusiones, consultas en la base de datos.

– Conexion: La clase responsable por la conexión con la base de datos.

– InicializaBD: Esta clase inicia la base de datos.

Hay que tener un elemento de la clase Conexión y un de la clase ComandosSql que son utilizados para la comunicación con la base de datos.

– ReiniciaBD: Esta clase reinicia la base de datos. Hay que tener un elemento de la clase Conexión y un de la clase ComandosSql que son utilizados para la comunicación con la base de datos.

– Diagnostico: La clase que haz la análisis de los artefactos y dice si ha alcanzado o no el nivel de madurez.

Esta clase hay que tener un elemento de la clase Conexión y un de la clase ComandosSql pues ella mira lo que hay en la base de datos y analiza si el nivel fue alcanzado.

Para dejar el sistema independiente del ordenador utilizado o de una base de datos inicial, fueran criadas las clases InicializaBD y ReiniciaBD. Estas clases crean una nueva base de datos con los proyectos sin ningún artefacto, sería la base de datos que el auditor debería tener al empezar una auditoría.

4.8 Interfaces

4.8.1 Interfaz Principal

La interfaz principal del programa contiene una pestaña para las Practicas Genéricas y una para las Practicas Específicas.

Hay también un botón llamado “DIAGNÓSTICO”, que a cualquier momento, el auditor poder pinchar para ver el resultado de la evaluación de los proyectos.

Sigue abajo la plantilla de la interfaz principal con la visión de la pestaña que contiene las prácticas genéricas.

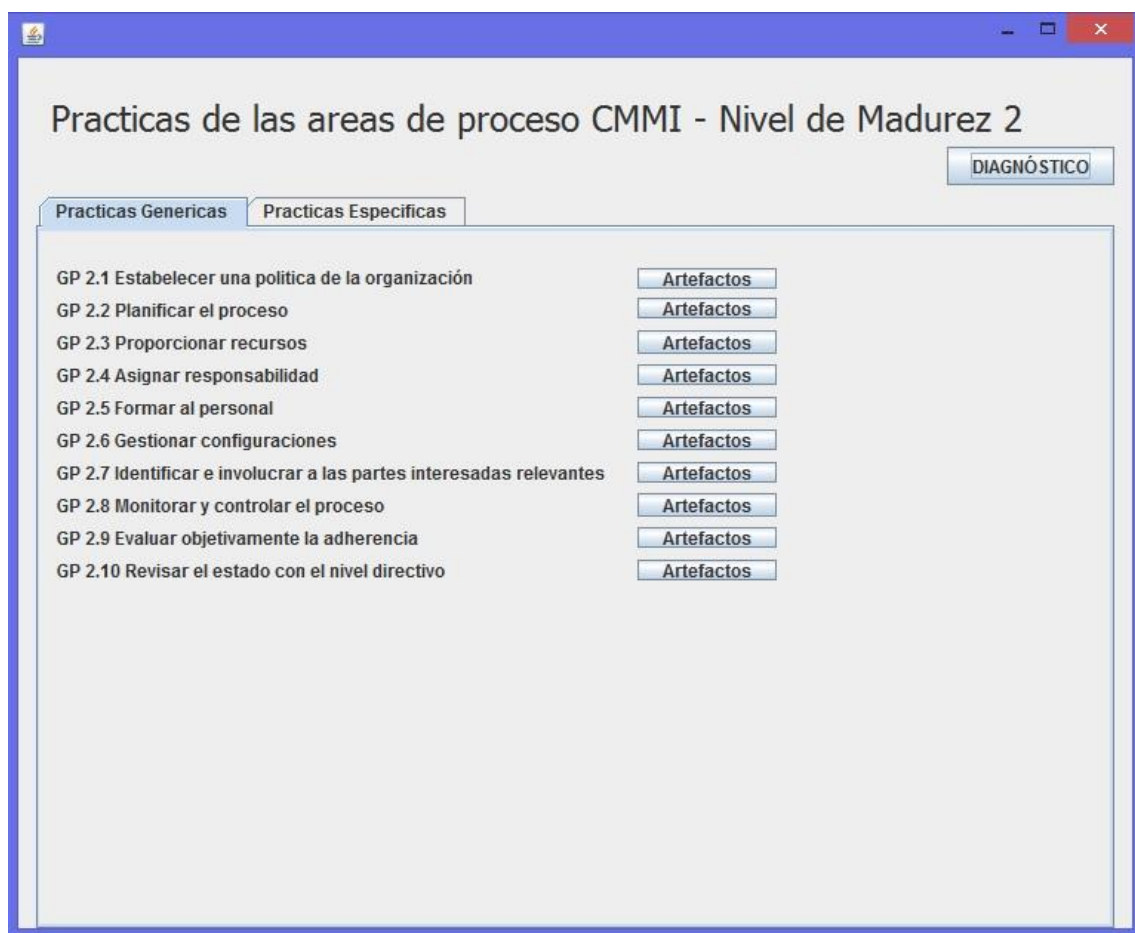


Ilustración 6: Interfaz Principal Prácticas Genéricas

Se nota que las prácticas genéricas de la pantalla son las mismas que están en la tablaPG citada en la sección “Base de Datos” y al lado de cada práctica hay un botón para el auditor insertar artefactos.

Sigue abajo la plantilla de la interfaz principal con la visión de la pestaña que contiene las prácticas específicas.

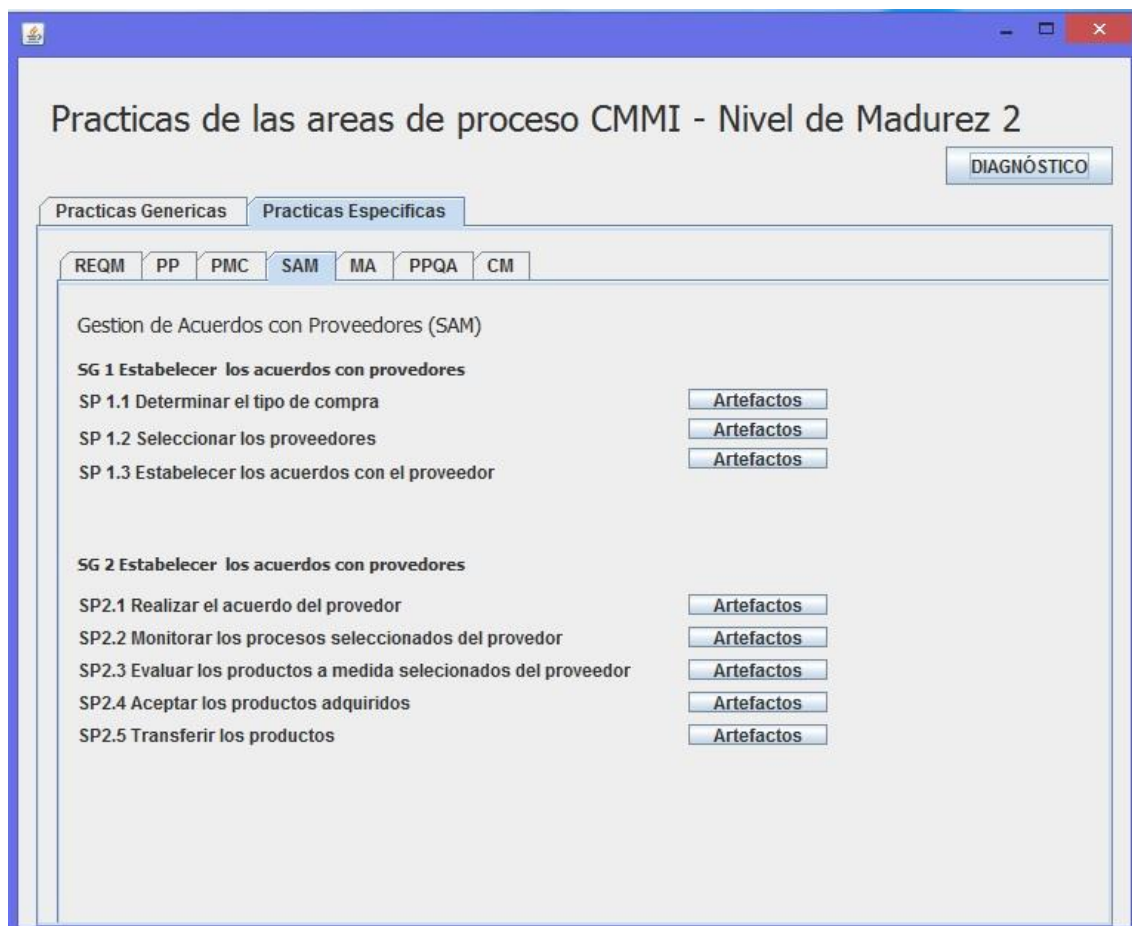


Ilustración 7: Interfaz Principal Prácticas Específicas

Se nota que hay un panel con 7 pestañas, donde cada una es una de las áreas de proceso necesarias para alcanzar el nivel de madurez 2.

Cada pestaña se refiere a una de las tablas citadas en la sección “Base de Datos” y al lado de cada práctica hay un botón para el auditor insertar artefactos. La pestaña REQM contiene las prácticas de la tablaREQM, la pestaña PP contiene las prácticas de la tablaPP, y así por delante.

4.8.2 Caja de Dialogo de los artefactos

Cuando el auditor pincha un botón “Artefactos”, surge una caja de dialogo como la siguiente:

PLANIFICACION DEL PROYECTO
GP2.1 Establecer una política de la organización

DIAGNÓSTICO

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO	AFIRMACION
PROYECTO 1	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia	Comunicación del plan de calidad a la empresa (correos, wiki, pagina Web)	
Link			
			Guardar
PROYECTO 2	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia	Comunicación del plan de calidad a la empresa (correos, wiki, pagina Web)	
Link			
			Guardar

Ilustración 8: Caja de Diálogo

Esta caja de dialogo contiene los datos del PIIDB de las tablas citadas en la sección “Base de Datos” para un determinado proceso.

La interfaz contiene primero, cual es la practica en cuestión. Después, contiene labels con los datos de artefacto directo y artefacto indirecto del PIIDB para esta práctica.

Para el JDialog de las prácticas específicas, cuya prácticas es la GP 2.1” Establecer una política de la organización”, hay el artefacto directo y el artefacto indirecto de la tablaGP cuyo ID es GP 2.1, y así por delante.

Además hay cajas de texto para dos proyectos (Proyecto 1 y proyecto 2), dónde el auditor inserta los artefactos encontrados y una caja de texto donde él puede poner, por ejemplo, el link de algún documento que cree que es relevante.

Al rellenar todo que quieras, el auditor debe pinchar el “Guardar”, si desea guardar las informaciones insertadas.

Cuando el auditor quiera hacer el diagnóstico él debe pinchar el botón “Diagnóstico”, entonces será analizado si el proyecto alcanzo el nivel deseado y enseñará la “interfaz Diagnóstico”.

4.8.3 Interfaz Diagnóstico

Para definir si la empresa ha alcanzado el nivel 2 de madurez, como se ha dicho anteriormente, utilizó la siguiente regla: Verificar si para todos los procesos hay una evidencia objetiva.

La definición de evidencia objetiva basada, como se ha dicho anteriormente, fue la que se encuentra en el Boletín citado y es la siguiente:

EVIDENCIA OBJETIVA = ARTEFACTO DIRECTO AND (ARTEFACTO INDIRECTO OR AFIRMACION)

Después de analizar si cada proyecto alcanzó el nivel de madurez, el software enseña el diagnóstico de los proyectos como en el siguiente dibujo de la interfaz:

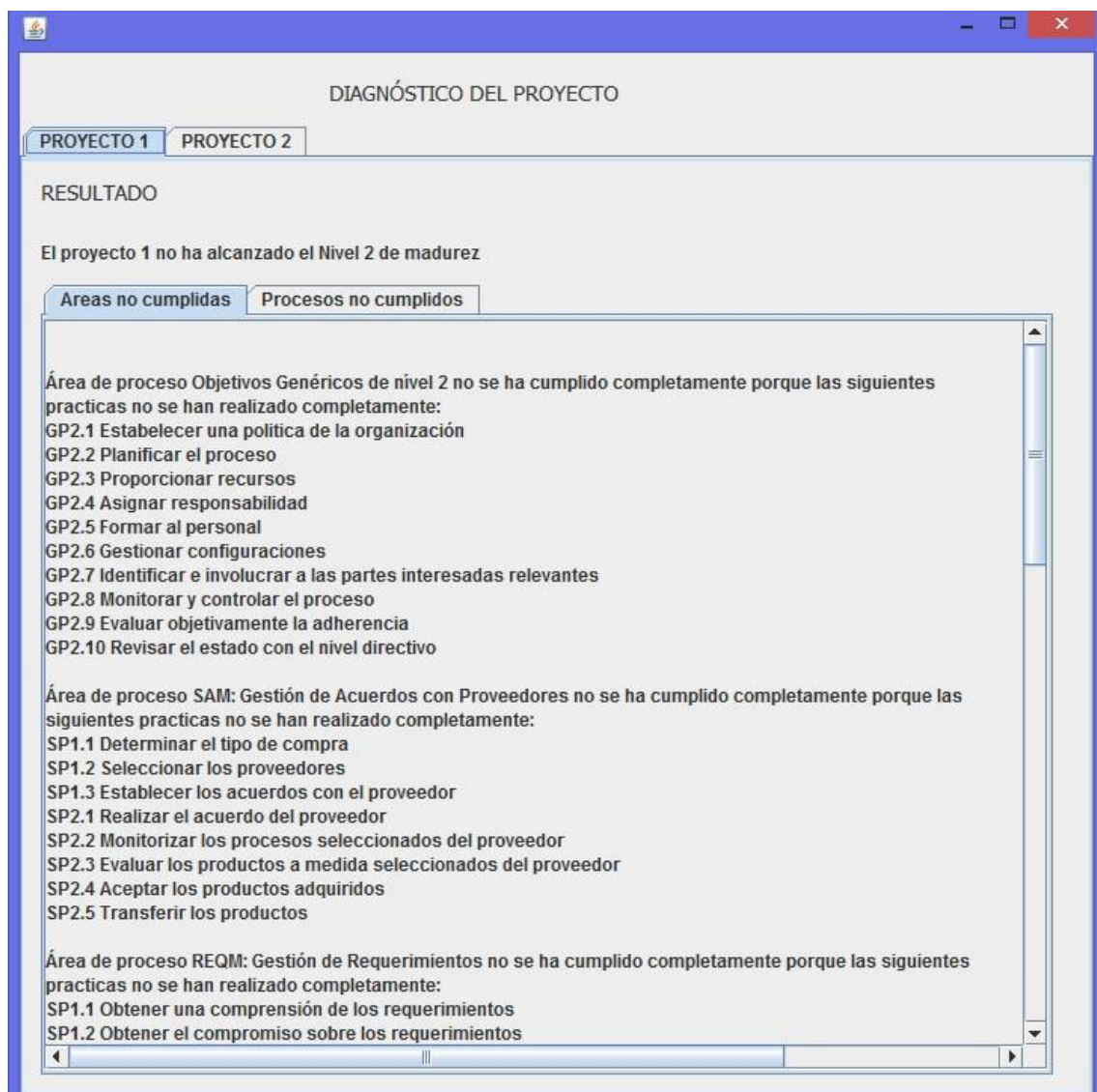


Ilustración 9: Interfaz del Diagnóstico con las áreas no cumplidas

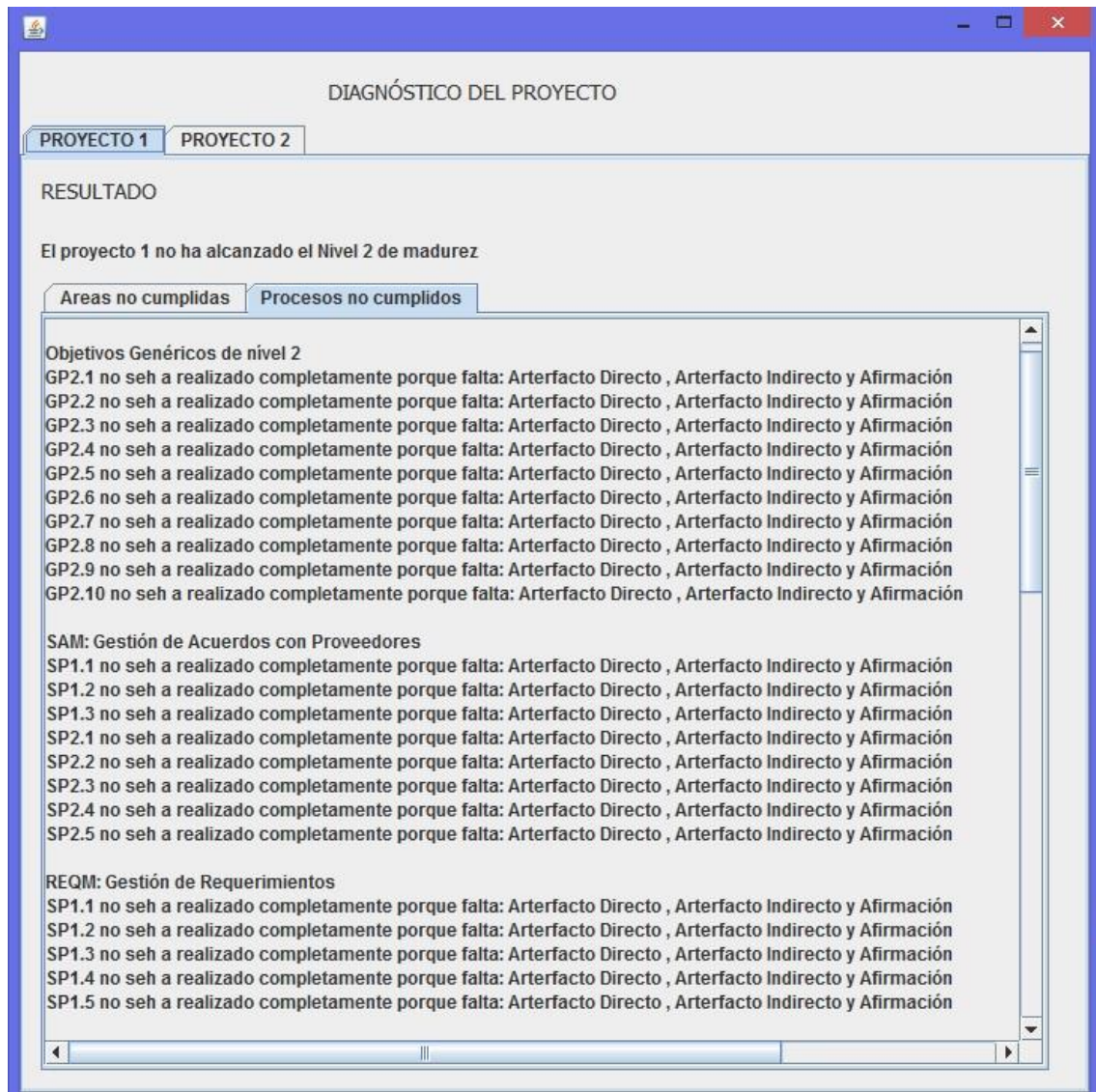


Ilustración 10: Interfaz del Diagnóstico con los procesos no cumplidos

Se nota que es una interfaz que contiene dos pestañas: una para cada proyecto. En cada pestaña hay un *label* que enseña el resultado al usuario, es decir, puede tener dos textos diferentes para el proyecto X, dónde X es 1 o 2: “El proyecto X ha alcanzado el nivel 2 de madurez” o “El proyecto X no ha alcanzado el nivel 2 de madurez”.

Después hay dos pestañas que sólo sirven para el segundo caso, pues contiene las áreas y los respectivos procesos que no fueron alcanzados.

4.9 Ejemplos Ejecuciones

4.9.1 Primera Ejecución

En la primera ejecución del sistema no va a haber ningún dato en la base de datos, entonces no va a haber ningún artefacto también. Sigue abajo las interfaces del proyecto en esta situación.

Al ejecutar el sistema, aparece la interface inicial que nunca va a cambiar pues no contiene datos que cambian, entonces va a quedar como la que fue enseñada anteriormente.

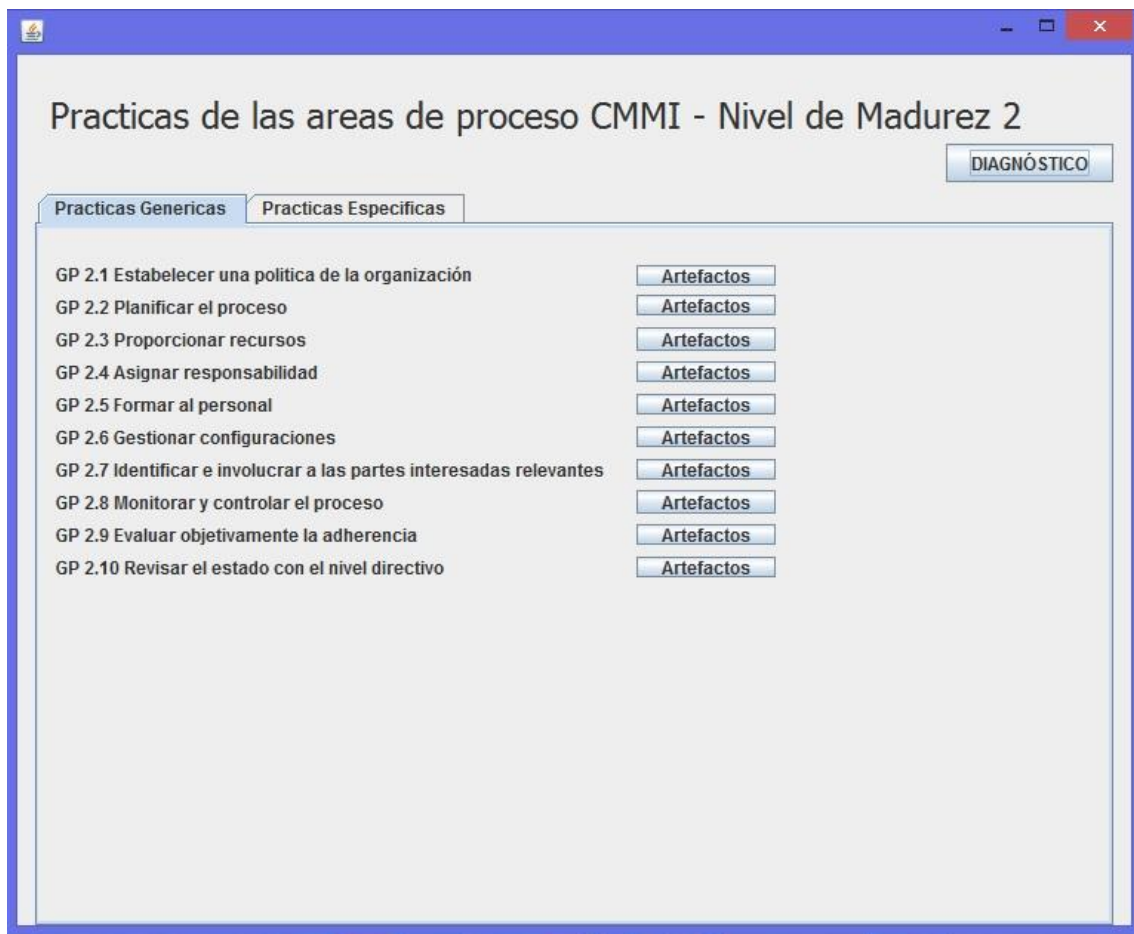


Ilustración 11: Interfaz Inicial – Primera Ejecución

Al pinchar el primero botón “Artefactos” del GP 2.1” Establecer una política de la organización”, surge una caja de dialogo como esa:

PLANIFICACION DEL PROYECTO
GP2.1 Establecer una política de la organización

DIAGNOSTICO

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO	AFIRMACION
PROYECTO 1	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia	Comunicación del plan de calidad a la empresa (correos, wiki, pagina Web)	
Link			
			Guardar
PROYECTO 2	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia	Comunicación del plan de calidad a la empresa (correos, wiki, pagina Web)	
Link			
			Guardar

Ilustración 12: Caja Dialogo – Primera Ejecución

Se inserta un artefacto directo como por ejemplo: “La empresa X tiene un plan de calidad de acuerdo con los requisitos necesarios.”, como en la figura:

PLANIFICACION DEL PROYECTO
GP2.1 Establecer una política de la organización

DIAGNOSTICO

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO	AFIRMACION
PROYECTO 1	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia	Comunicación del plan de calidad a la empresa (correos, wiki, pagina Web)	
Link			
			Guardar
PROYECTO 2	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia	Comunicación del plan de calidad a la empresa (correos, wiki, pagina Web)	
Link			
			Guardar

Ilustración 13: Caja Dialogo – Primera Ejecución, Inserta artefacto directo

Después de pinchar el guardar, se pincha el Diagnóstico, entonces la pantalla “Diagnóstico” se queda como se sigue:

The screenshot shows two windows. The left window, 'PLANIFICACION DEL PROYECTO', displays 'GP2.1 Establecer una política de la organización' with a description of a quality plan and a 'Link' field. The right window, 'DIAGNÓSTICO DEL PROYECTO', shows 'PROYECTO 1' selected. Under 'RESULTADO', it states 'El proyecto 1 no ha alcanzado el Nivel 2 de madurez'. The 'Áreas no cumplidas' tab is active, listing three areas: 'Objetivos Genéricos de nivel 2', 'SAM: Gestión de Acuerdos con Proveedores', and 'REQM: Gestión de Requerimientos'. Each area lists specific practices (GP2.1 to GP2.10, SP1.1 to SP2.5) that have not been fully completed.

Ilustración 14: Diagnóstico de la Primera Ejecución, Áreas no cumplidas

Se nota que el área GP 2.1” Establecer una política de la organización” no se ha cumplido.

This screenshot shows the same 'DIAGNÓSTICO DEL PROYECTO' window, but with the 'Procesos no cumplidos' tab selected. It provides a detailed breakdown of why specific processes were not completed. For example, under 'Objetivos Genéricos de nivel 2', it lists GP2.1 through GP2.10, each followed by the reason: 'no se ha realizado completamente porque falta: Arterfacto Indirecto y Afirmación'. Similar details are provided for the 'SAM' and 'REQM' process groups.

Ilustración 15: Diagnóstico de la Primera Ejecución, Procesos no cumplidos

Se nota que el área GP 2.1” Establecer una política de la organización” no se ha cumplido porque no hay ni artefacto indirecto, ni afirmación.

Se inserta un artefacto indirecto como en la ilustración abajo. Supongamos que La comunicación es hecha por wiki.

PLANIFICACION DEL PROYECTO
GP2.1 Establecer una política de la organización

DIAGNÓSTICO

	ARTEFACTO DIRECTO	ARTEFACTO INDIRECTO	AFIRMACION
PROYECTO 1	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia	La empresa X tiene un plan de calidad de acuerdo con los requisitos necesarios.	Comunicación del plan de calidad a la empresa (correos, wiki, pagina Web)
Link			
			Guardar
PROYECTO 2	Plan de calidad que contemple el desarrollo software, los procesos de nivel 2 y que se encuentre firmado y respaldado por la gerencia		
Link			
			Guardar

Ilustración 16: Caja Dialogo – Primera Ejecución, Inserta artefacto indirecto

Después de guardar el nuevo artefacto y pedir el diagnóstico, la pantalla se queda como:

DIAGNÓSTICO DEL PROYECTO

PROYECTO 1 PROYECTO 2

RESULTADO

El proyecto 1 no ha alcanzado el Nivel 2 de madurez

Áreas no cumplidas Procesos no cumplidos

Área de proceso Objetivos Genéricos de nivel 2 no se ha cumplido completamente porque las siguientes practicas no se han realizado completamente:

- GP2.2 Planificar el proceso
- GP2.3 Proporcionar recursos
- GP2.4 Asignar responsabilidad
- GP2.5 Formar al personal
- GP2.6 Gestionar configuraciones
- GP2.7 Identificar e involucrar a las partes interesadas relevantes
- GP2.8 Monitorizar y controlar el proceso
- GP2.9 Evaluar objetivamente la adherencia
- GP2.10 Revisar el estado con el nivel directivo

Área de proceso SAM: Gestión de Acuerdos con Proveedores no se ha cumplido completamente porque las siguientes practicas no se han realizado completamente:

- SP1.1 Determinar el tipo de compra
- SP1.2 Seleccionar los proveedores
- SP1.3 Establecer los acuerdos con el proveedor
- SP2.1 Realizar el acuerdo del proveedor
- SP2.2 Monitorizar los procesos seleccionados del proveedor
- SP2.3 Evaluar los productos a medida seleccionados del proveedor
- SP2.4 Aceptar los productos adquiridos
- SP2.5 Transferir los productos

Área de proceso REQ: Gestión de Requerimientos no se ha cumplido completamente porque las siguientes practicas no se han realizado completamente:

- SP1.1 Obtener una comprensión de los requerimientos
- SP1.2 Obtener el compromiso sobre los requerimientos
- SP1.3 Gestionar los cambios en los requerimientos

Ilustración 17: Diagnóstico de la Primera Ejecución, Áreas no cumplidas, después de inserción de evidencia

Se nota que el Área de proceso Objetivo Genérico todavía no se ha cumplido, pero el proceso GP 2.1” Establecer una política de la organización” no está en la lista de procesos del área no cumplidos, es decir el proceso GP 2.1 se ha cumplido.

El mismo vale para la lista de procesos no cumplidos.

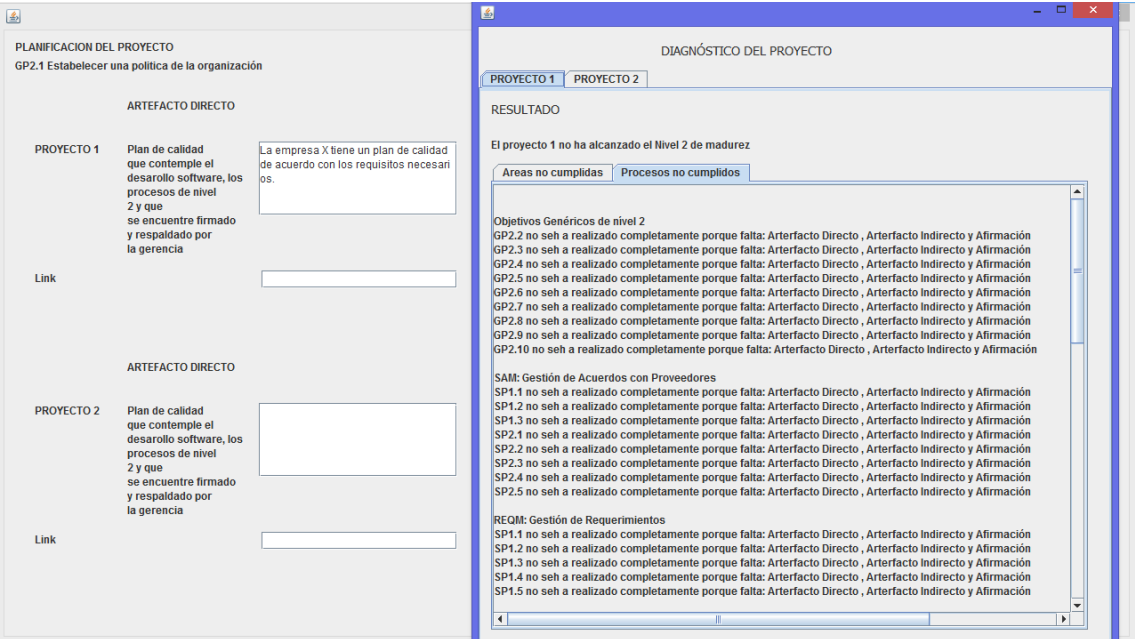


Ilustración 18: Diagnóstico de la Primera Ejecución, Procesos no cumplidos, después de inserción de evidencia

Se nota que el proceso GP 2.1” Establecer una política de la organización” no está en la lista de procesos no cumplidos, es decir el proceso GP 2.1 se ha cumplido.

4.9.2 Ejecución con evidencia en casi todas las áreas

Sigue abajo un caso donde el auditor ha encontrado evidencia para todos los procesos con excepción de la Practica Específica “SP 2.1 Comunicar y asegurar la resolución de las no-conformidades”.

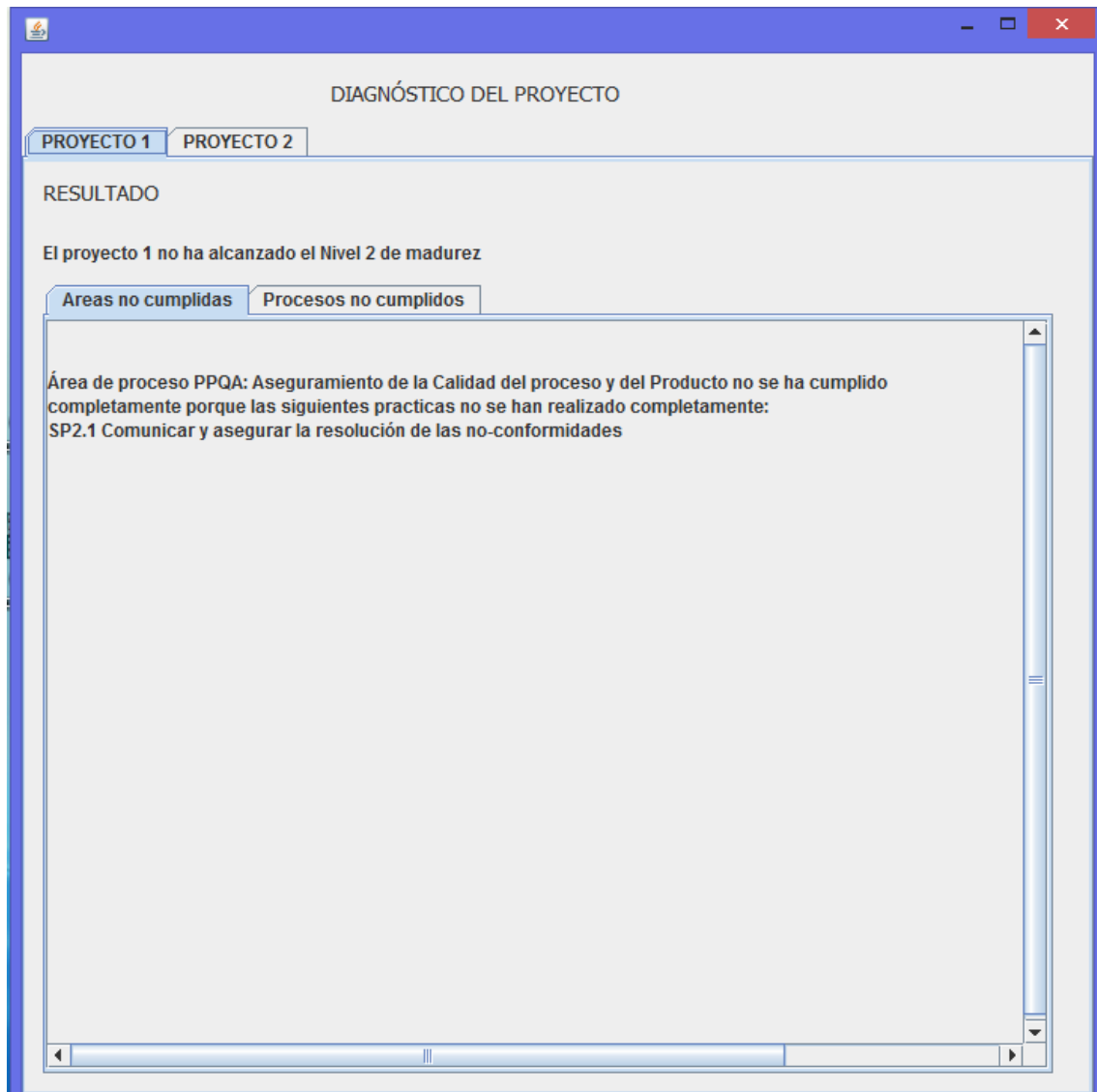


Ilustración 19: Pantalla Diagnóstico con las áreas no cumplidas para el caso citado.

En la pantalla está que el Resultado del diagnóstico es que el proyecto no ha alcanzado el nivel 2 de madurez y luego abajo se explica por qué: en el área PPQA (Aseguramiento de la calidad de proceso y producto) no hay una evidencia en la Practica Específica “SP 2.1 Comunicar y asegurar la resolución de las no-conformidades”.

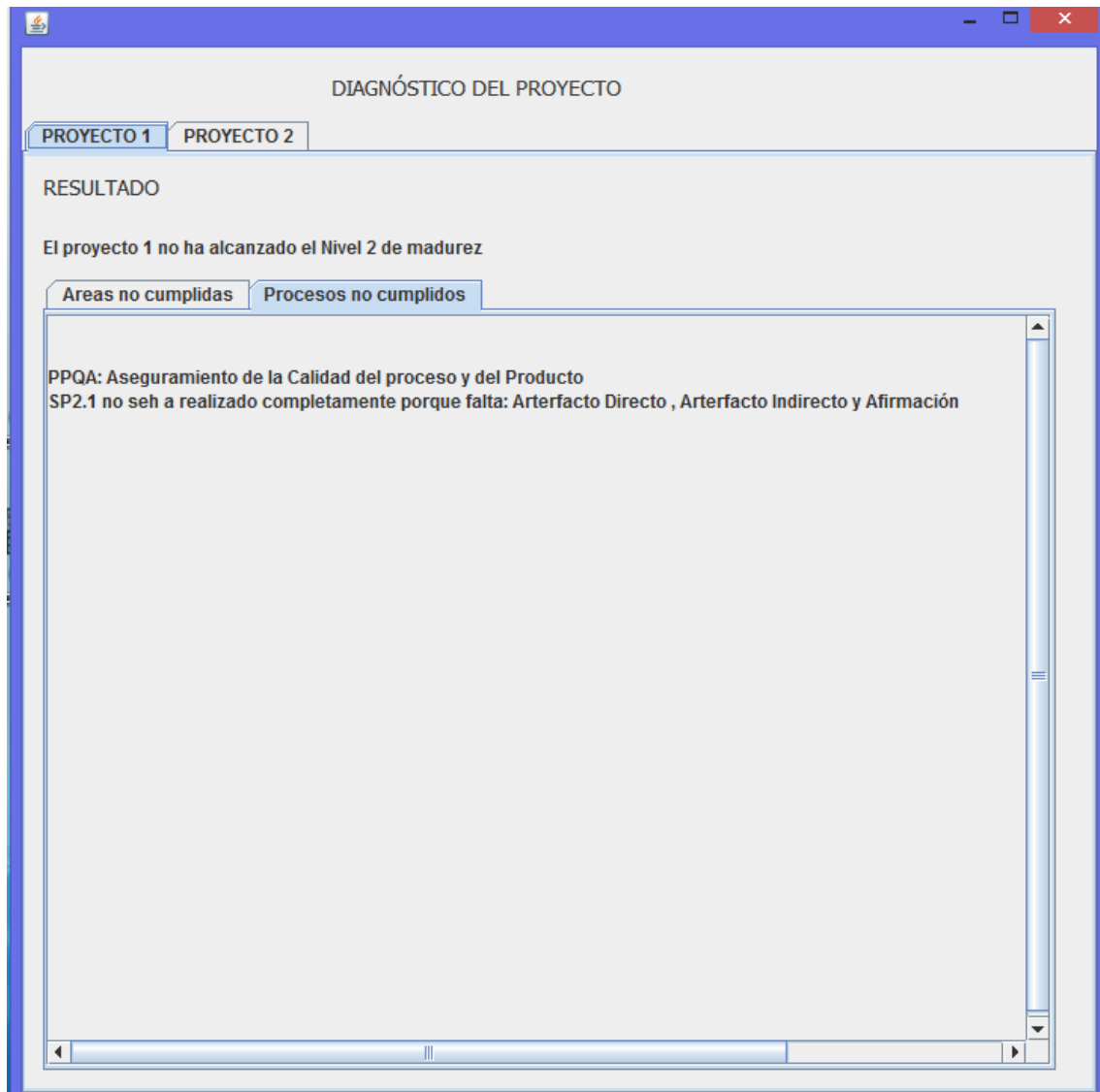


Ilustración 20: Pantalla Diagnóstico con los procesos no cumplidos para el caso citado.

La pestaña “procesos no cumplidos” especifica lo que la pestaña Áreas no cumplidas había dicho. La Practica Específica “SP 2.1 Comunicar y asegurar la resolución de las no-conformidades” no contiene una evidencia porque no hay artefacto directo, ni artefacto indirecto, ni afirmación. Para que tuviera evidencia debería tener un artefacto directo y un artefacto indirecto y/o una afirmación.

Después de insertar la evidencia que quedaba, el diagnóstico se quedó como se sigue:

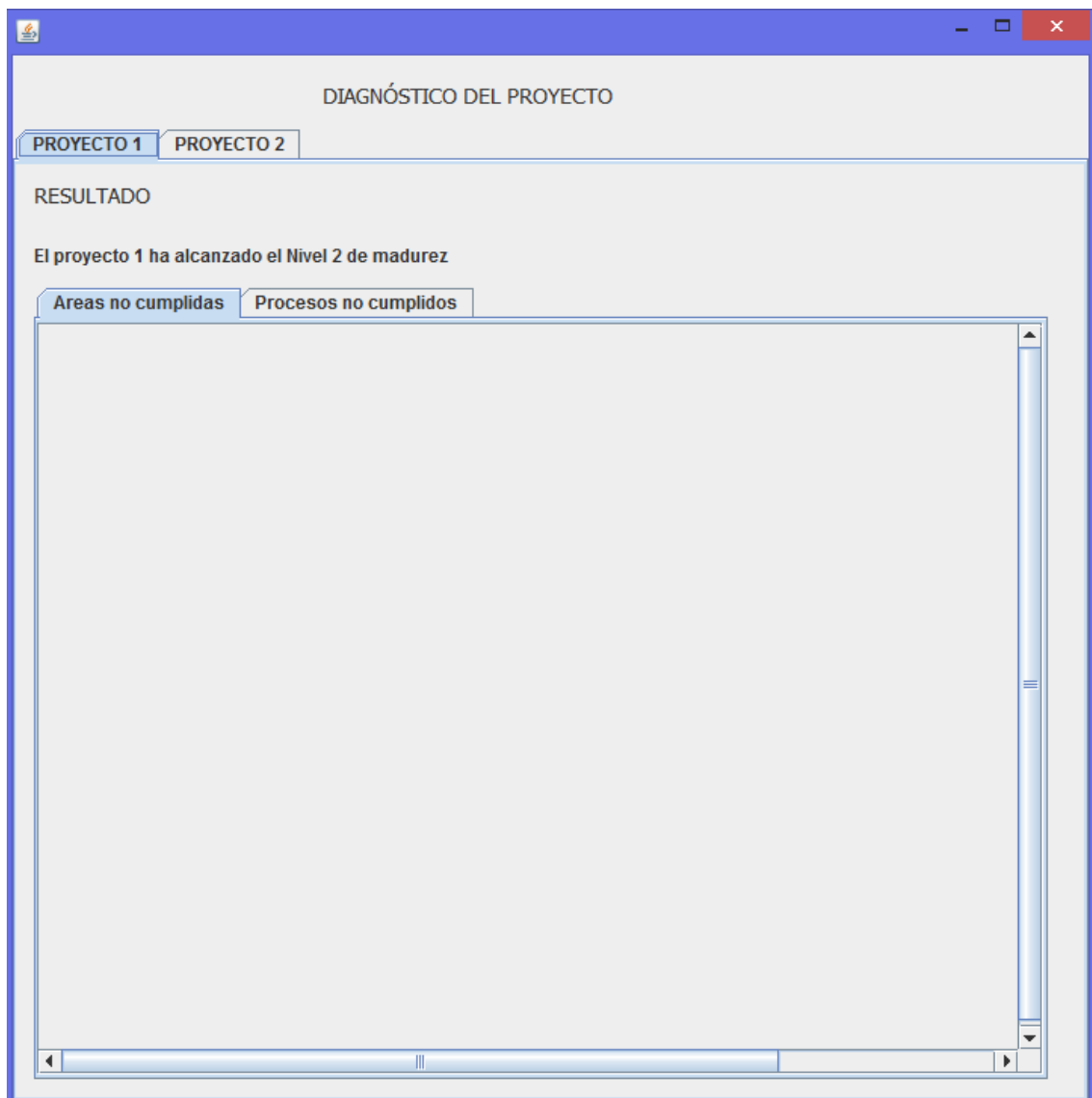


Ilustración 21: Caso dónde se alcanzó el nivel 2 de madurez

Ahora, se nota que el proyecto ha alcanzado el nivel 2 de madurez.

5 Conclusiones y trabajo futuro

Los objetivos propuestos para ese proyecto han sido cubiertos. Fue desarrollado y probado un sistema que atiende todos los requisitos definidos.

El desarrollo fue muy interesante, pues fue una oportunidad de poner en práctica muchos conceptos aprendidos en clase y la experiencia de recibir un problema, planear su solución, buscar las herramientas más adecuadas a su implantación, desarrollar un sistema que atiende los requisitos definidos y lo probar con éxito, sobretodo sólo, seguramente es útil para distintos momentos de la carrera de un ingeniero informático.

Este proyecto se puede extender y mejorar en diversos aspectos.

Primeramente, hay que probarlo en un caso real y con el feedback del usuario adaptar el software a su uso. Para auxiliar a la empresa, se puede hacer un guía para que la empresa sepa lo que hay que hacer para alcanzar el nivel 2 de madurez.

Después, hay que definir los artefactos directos, artefactos indirectos y afirmaciones de las prácticas genéricas para todas las áreas de proceso y poner en el alcance del diagnóstico.

Otra ampliación interesante sería vincularlo a una base de datos documental, entonces el software sería más completo.

Además, el proyecto sólo sirve al caso de diagnóstico de nivel 2, entonces hay que ampliarlo a los restos de niveles del CMMI.

Bibliografía

- [1] S. E. I. Carnegie Melon University, *Capability Maturity Model Integration (CMMI), Version 1.1. CMMI for Software Engineering. Technical Report # CMU/SEI-2001-TR-029.*, August, 2002..
- [2] C. institute, “ <http://cmmiinstitute.com/>,” [Online].
- [3] E. T. S. d. I. I. U. R. J. Carlos, “*Guía práctica de supervivencia en una auditoria CMMI* “(*Numero 2011-002*), Boletín de la ETSII ISSN: 2172-6620.
- [4] “Certificación CMMI-SW - Capítulo 4,” em *Certificación CMMI-SW*, pp. http://sedici.unlp.edu.ar/bitstream/handle/10915/3956/4_-_Certificaci%C3%B3n_CMMI-SW.pdf?sequence=9.
- [5] SEI, “CMMI para Desarrollo, Versión - 1.3,” [Online]. Available: <http://www.sei.cmu.edu/library/assets/whitepapers/Spanish%20Technical%20Report%20CMMI%20V%201%203.pdf>.
- [6] E. d. i. d. s. Vates. [Online]. Available: <http://www.vates.com/cmml/cmml.html>.

Anexo 1: Instalación

Junto al código hay dos ejecutables .jar para el caso donde se desea sólo instalar y ejecutar la aplicación.

Para esto es necesario primero que el ordenador dónde el aplicativo va a ser instalado tenga previamente instalado un servicio de MySQL de la versión 5.5, el MySQL Community Server, que puede ser bajado de la página del MySQL: <http://www.mysql.com/>.

Después hay que crear en una base de datos cuyo usuario es “root” y la contraseña “123456” una base de datos vacía llamada “base”, que va a ser la base usada por el aplicativo. Para facilitar la creación de la base de datos se puede instalar el aplicativo “MySQL Workbench” que aparece el siguiente nombre en la página de MYSQL: mysql-workbench-gpl-5.2.47-win32.msi. Después que el banco de datos esté configurado, hay que ejecutar primero el aplicativo AuditoriaCMMIInicia.jar que es un ejecutable que va a inicializar la base de datos “base” con los datos básicos de los proyectos y tablas. Ese aplicativo sólo hay que ejecutar la primera vez, pues él va a limpiar la base de datos, a menos que el usuario quiera limpiar la base de datos. Él se encuentra en la carpeta “AuditoriaCMMI\ejecutables”.

Con la base inicial lista ya se puede ejecutar el AuditoriaCMMI.jar, que es el aplicativo desarrollado y que tienes todas las características citadas en esta memoria. Él se encuentra en la carpeta “AuditoriaCMMI\ejecutables” también.

Para ejecutar un ejecutable fichero.jar en el sistema operacional Windows:

- Abrir el “prompt de comandos”
- Elegir el directorio donde está el fichero: `cd C:\Directorio`
- Escribir el comando: `“java -jar fichero.jar”`

Anexo2: Ejecución en Eclipse con MySQL

Para la compilación en eclipse es necesario primero, bajar la versión más reciente del Eclipse en la página: <http://www.eclipse.org/>

Después hay que bajar el “Java Development kit”, en este proyecto fue usado el JDK 7, y configurar el PATH del Windows. En este caso fue configurado las variables de ambiente del Windows como:

Crio una nueva variable “ClassPath” con el valor: “;C:\Program Files\Java\jdk1.7.0_17\src.zip;C:\Program Files\Java\jdk1.7.0_17\lib\tools.jar;”

Añadió en la variable Path: “C:\Program Files\Java\jdk1.7.0_17\bin”

Para la base de datos, fue instalado un servicio de MySQL de la versión 5.5, el MySQL Community Server, que puede ser bajado de la página del MySQL: <http://www.mysql.com/>. Para facilitar la creación de la base de datos fue instalado el aplicativo “MySQL Workbench” que aparece el siguiente nombre en la página de MYSQL: mysql-workbench-gpl-5.2.47-win32.msi. Se creó en una base de datos cuyo usuario es “root” y la contraseña “123456” una base de datos vacía llamada “base”, que es la base usada por el aplicativo. Para permitir la conversación entre MySQL y Eclipse hay que bajar también un Conector JAVA llamado mysql-connector-java-5.1.24- bin, que se encuentra en la pasta “Lib” del código.

Después de preparar el ambiente de desarrollo, basta importar el proyecto. Es importante notar que en la clase Main hay el siguiente código:

```
//ReiniciaBD reinicia = new ReiniciaBD();  
  
MainFrame frame = new MainFrame();  
frame.setVisible(true);
```

En el Main se puede elegir si quiere reiniciar la base de datos cuando ejecutar el aplicativo y con eso limpiar todos los datos insertados anteriormente o si prefiere mantener los datos en la base de datos.

Para generar los ejecutables .jar citados, sólo se ha cambiado el Main. Primero se generó una versión donde reinicia la base datos, el, y después una que mantiene los datos